



CARLOS ALBERTO PÚPIO DE OLIVEIRA

**USO DA FUNÇÃO DE DIRICHLET EM TESTES
DE HIPÓTESE DA ALEATORIEDADE
ESPACIAL EM CONFIGURAÇÕES PONTUAIS**

**LAVRAS – MG
2010**

CARLOS ALBERTO PÚPIO DE OLIVEIRA

**USO DA FUNÇÃO DE DIRICHLET EM TESTES DE HIPÓTESE DA
ALEATORIEDADE ESPACIAL EM CONFIGURAÇÕES PONTUAIS**

Dissertação apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-Graduação em Estatística e Experimentação Agropecuária, área de concentração em Estatística e Experimentação Agropecuária, para a obtenção do título de Mestre.

Orientador

Dr. João Domingos Scalon

**LAVRAS - MG
2010**

**Ficha Catalográfica Preparada pela Divisão de Processos Técnicos da
Biblioteca da UFLA**

Oliveira, Carlos Alberto Púpio de.

Uso da função de Dirichlet em testes de hipótese da
aleatoriedade espacial em configurações pontuais / Carlos Alberto
Púpio de Oliveira. – Lavras : UFLA, 2010.

94 p. : il.

Dissertação (mestrado) – Universidade Federal de Lavras, 2010.

Orientador: João Domingos Scalon.

Bibliografia.

1. Função K de Ripley. 2. Estatística espacial. 3. Agrupamento.
I. Universidade Federal de Lavras. II. Título.

CDD – 519.56

CARLOS ALBERTO PÚPIO DE OLIVEIRA

**USO DA FUNÇÃO DE DIRICHLET EM TESTES DE HIPÓTESE DA
ALEATORIEDADE ESPACIAL EM CONFIGURAÇÕES PONTUAIS**

Dissertação apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-Graduação em Estatística e Experimentação Agropecuária, área de concentração em Estatística e Experimentação Agropecuária, para a obtenção do título de Mestre.

APROVADA em 03 de setembro de 2010.

Dr^a. Gláucia Amorim Faria UFRB

Dr. Renato Ribeiro de Lima UFLA

Dr. João Domingos Scalon

Orientador

LAVRAS - MG

2010

A Deus, por ter me dado forças nos momentos difíceis.

A minha mãe Mafalda Púpio (*in memoriam*), pelos ensinamentos.

Aos meus irmãos, Altamira, Rosemira, José Augusto, Maria de Nazaré,
Vera Lúcia, João e Amadeu, pelo apoio.

A minha esposa Nazira e aos meus filhos, Andréa, Abrahão, Vítor,
Vinícius e Vitória Maria, pela confiança,

Aos meus netos, Eduardo, Bárbara e Ana Clara, pelo florescimento de
um novo momento.

DEDICO

AGRADECIMENTOS

A Deus, que sempre foi justo na minha vida, não deixando o meu caminho na escuridão.

À Universidade Federal do Acre (UFAC), pela minha liberação.

A todo o corpo docente do Colégio de Aplicação e a todos os servidores administrativos que acreditaram no meu sucesso.

À Universidade Federal de Lavras (UFLA), em especial ao Departamento de Ciências Exatas (DEX), pela realização deste trabalho.

Aos professores do DEX, pelos valiosos ensinamentos.

Ao meu orientador, João Domingos Scalon, pela grande atenção, competência, dedicação e arte de ensinar.

A todos os colegas de pós-graduação em Estatística e Experimentação Agropecuária. Em especial, ao Crysttian, Hernani e Ademir, pelos grandes momentos de estudos. Ao Moisés, Paulo Emiliano e Fábio, pela ajuda no R.

Às meninas, Josi (P e G), Maria, Selminha e Dilinha, por tudo.

A minha querida mãe (*in memorian*), pelos ensinamentos, dedicação e carinho.

Aos meus irmãos, cunhados (as), sobrinhos (as), primos (as), pela força.

Aos meus filhos, minha fonte de energia.

Aos meus netos, pelo nascimento de uma nova esperança.

A minha esposa Nazira, pelo apoio, amizade e compreensão.

Aos colegas acrianos, Eustáquio, Ed Carlos, Naje e Altemir, pelo espírito de luta de cada acriano.

RESUMO

Uma configuração espacial de pontos é um conjunto de localizações pontuais distribuídas por um processo estocástico dentro de alguma região. A análise estatística de dados de uma configuração espacial de pontos normalmente começa com um teste contra a hipótese de aleatoriedade espacial. O objetivo do presente trabalho é investigar o desempenho de novos testes para aleatoriedade espacial que tem como base as áreas de um mosaico de Dirichlet (H). Os novos testes são comparados com o teste baseado na função K de Ripley. Os testes baseados nas funções K e H são aplicados em um grande número de configurações com agrupamentos e regularidade gerados através de simulação computacional. O número de vezes que o teste detecta não aleatoriedade em configurações não aleatórias fornece uma indicação do seu poder. Os resultados mostraram que as funções K e H apresentam o mesmo poder para configurações com forte afastamento da hipótese nula de completa aleatoriedade espacial. Entretanto, a função K é mais poderosa que a função H para configurações próximas da completa aleatoriedade espacial. Finalmente, os testes baseados nas funções K e H foram aplicados a um conjunto real de dados e chegaram às mesmas conclusões.

Palavras-chave: Simulação Monte Carlo. Configuração de pontos. Estatística espacial. Função K . Testes de hipóteses. Floresta.

ABSTRACT

A spatial point pattern is a set of point locations distributed by a stochastic process within some region. The statistical analysis of spatial point pattern data usually begins with a test for spatial randomness. The aim of the present work is to investigate the performance of new tests for spatial randomness based on Dirichlet tessellations (H). The new tests are compared to tests based on the K function. The tests based on the K and H functions are applied in a large number of patterns from cluster and regularity processes generated via computer simulation. The number of times that the test successfully detects non-randomness in non-random patterns gives an indication of its power. The results showed that both K and H functions presented the same power for patterns with strong evidences against completely spatial randomness. However, the K function appears to be more powerful than the H function for patterns close to the complete spatial randomness. Finally, the tests based on both K and H functions were applied to an actual data set and led to the same conclusions.

Keywords: Monte Carlo simulation. Point patterns. Spatial statistics. K -function. Hypotheses tests. Forestry.

LISTA DE FIGURAS

Figura 1	Três padrões de pontos simulados mostrando realizações com completa aleatoriedade espacial, agrupamento e regularidade.....	15
Figura 2	Mapa do Dr. John Snow mostrando agrupamento de óbitos por cólera na cidade de Londres.....	16
Figura 3	Realização de um processo pontual com agrupamento com 100 pontos, 4 grupos e variabilidade radial de 0,05...	21
Figura 4	Realização de um processo pontual, com regularidade, com 200 pontos.....	22
Figura 5	Ideia gráfica de uma função K	27
Figura 6	Mosaico de Dirichlet com 42 pontos.....	30
Figura 7	Mosaico de Dirichlet com 15 pontos.....	34
Figura 8	Simulação de um processo de regularidade com 100 pontos e distância radial de 0,08, usando a função K de Ripley e função de Dirichlet.....	39
Figura 9	Simulação de um processo de agrupamento com 100 pontos, 10 grupos com 10m pontos cada, variabilidade radial de 0,05, usando as funções K de Ripley e Dirichlet	42
Figura 10	Gráfico da função de poder.....	43
Figura 11	Gráfico do poder do teste contra regularidade.....	51
Figura 12	Gráfico do poder do teste contra agrupamento com 2 grupos.....	52
Figura 13	Gráfico do poder do teste contra agrupamento com 5 grupos.....	53
Figura 14	Gráfico do poder do teste contra agrupamento com 10 grupos.....	54

Figura 15	Gráfico do poder do teste contra agrupamento com 20 grupos.....	56
Figura 16	Plotagem das arvores de copaíba.....	58
Figura 17	Gráfico do envelope de simulação da função K de Ripley	59
Figura 18	Gráfico do envelope de simulação da função de Dirichlet	60

LISTA DE TABELAS

Tabela 1	Coordenadas cartesianas (x , y) de cada árvore de copaíba no fragmento florestal	46
Tabela 2	Número de vezes em que os testes baseados nas funções K e H identificaram corretamente a configuração com agrupamentos (poder) em 1000 realizações.....	50
Tabela 3	Número de vezes em que os testes baseados nas funções K e H identificaram corretamente a configuração com regularidade (poder) em 1000 realizações.....	55

SUMÁRIO

1	INTRODUÇÃO	12
2	REFERENCIAL TEÓRICO	13
2.1	Estatística Espacial	13
2.2	Processos Pontuais Espaciais	14
2.3	Completa Aleatoriedade Espacial (CSR)	16
2.4	Processos pontuais com Agrupamentos	19
2.5	Processo pontuais com regularidade	21
2.6	Descrição de processos pontuais espaciais	22
2.6.1	Função F	24
2.6.2	Função G	25
2.6.3	Função J	25
2.6.4	Função K de Ripley	26
2.6.5	Mosaico de Dirichlet	28
2.7	Teste de hipótese e poder	31
2.8	Análise de CSR usando a função K	34
2.9	Análise de CSR usando a função de Dirichlet	37
3	MATERIAL E MÉTODO	44
3.1	Simulação	44
3.2	Aplicação	45
4	RESULTADO E DISCUSSÃO	48
4.1	Simulação	48
4.2	Aplicação	56
5	CONCLUSÃO	62
	REFERENCIAS	63
	ANEXOS	66

1 INTRODUÇÃO

Muitos fenômenos são expressos através de ocorrências identificadas como pontos localizados no espaço, denominados processos pontuais espaciais. São exemplos: localização de crimes, ocorrências de doenças e localização de espécies vegetais. Observa-se que o objeto de interesse é a própria localização espacial dos eventos em estudo. O objetivo inicial de análise desses fenômenos é estudar a distribuição espacial dos eventos, testando hipótese de distribuição aleatória contra as hipóteses alternativas de agrupamentos ou regularidade.

A literatura apresenta diversos métodos para testar a hipótese de aleatoriedade espacial de eventos pontuais. O método mais simples é baseado na contagem do número de pontos por área. Outros métodos mais refinados são: vizinhos mais próximos baseados nas funções F , G e J e aqueles baseados na função K de Ripley. Muitos desses métodos podem ser generalizados para situações mais complexas como eventos localizados em três dimensões.

Alguns autores sugerem que a análise dos mosaicos (ou triângulos) de Dirichlet (ou Voronoi) pode ser utilizada para testar a hipótese de aleatoriedade espacial em uma configuração de pontos. O presente trabalho tem como objetivo propor uma análise gráfica e testes baseados nos mosaicos de Dirichlet, utilizando simulação de Monte Carlo. O poder dos testes propostos é comparado ao poder dos testes baseados na função K de Ripley.

Finalmente, os dois testes são aplicados a um caso real. Foram utilizados os dados de coordenadas de 20 árvores de copaíba, localizadas na reserva florestal da Universidade Federal de Lavras. A análise teve como objetivo verificar se as árvores de copaíba estão distribuídas de forma aleatória, agrupada ou regular na reserva florestal.

2 REFERENCIAL TEÓRICO

2.1 Estatística espacial

A análise estatística espacial pode ser definida como uma técnica que busca descrever os padrões existentes nos dados espaciais e estabelecer, preferencialmente, de forma quantitativa, os relacionamentos entre as diversas variáveis geográficas (RIPLEY, 1976). Em outras palavras, o que diferencia a estatística usual da estatística espacial é que, nesta última, a posição do dado é explicitamente incluída no modelo.

Questões complexas surgem na análise espacial, muitas das quais não são claramente definidas, nem totalmente resolvidas, mas constituem a base para a investigação em curso. Uma dessas questões é fundamental e se refere à problemática definição da localização espacial das entidades em estudo. Por exemplo, um estudo sobre a saúde humana poderia descrever a posição espacial dos seres humanos com um ponto colocado onde vivem, ou com um ponto localizado no local em que trabalham, ou usando uma linha para descrever suas viagens semanais. Cada uma dessas escolhas tem efeitos diretos sobre as técnicas que podem ser utilizadas para a análise e para as conclusões resultantes desse processo. Outras questões em análise espacial incluem as limitações do conhecimento matemático e a forma como os cálculos são feitos, os pressupostos exigidos pelas atuais técnicas estatísticas e problemas de computação (RIPLEY, 1976).

A classificação das técnicas de análise espacial é difícil devido ao grande número de diferentes áreas de pesquisa envolvidas, às diferentes abordagens fundamentais que podem ser escolhidas e às muitas formas que os dados podem assumir. A taxonomia mais utilizada para descrever o problema da análise espacial considera três tipos de dados conforme pode ser visto em Bailey

et al. (1995), Cressie (1991) e Câmara et al. (2002): processos pontuais, superfícies contínuas e áreas com contagens.

Processos pontuais: são fenômenos expressos através de ocorrências identificadas como pontos localizados no espaço, denominados processos pontuais. A localização da ocorrência de casos de doenças e a localização de indivíduos de uma determinada espécie são exemplos desse tipo de dados.

Superfícies contínuas (Geoestatística): são fenômenos que se distribuem continuamente em uma região. Usualmente, esse tipo de dado resulta de um levantamento de recursos naturais, incluindo mapas geológicos, topográficos e ecológicos. Um exemplo desse tipo de dados são medidas da concentração de um elemento químico no solo.

Áreas com contagens: são fenômenos associados aos dados de levantamentos populacionais, como censos e que originariamente se referem a indivíduos localizados em pontos específicos no espaço. Normalmente, esses dados são agregados em unidades de análises, usualmente delimitadas por polígonos fechados, tais como setores censitários, municípios e microrregiões.

2.2 Processos pontuais

A localização dos pontos e a relação entre eles é uma característica importante de um processo pontual, verificando se os eventos observados apresentam algum tipo de padrão sistemático, ao invés de estarem distribuídos aleatoriamente. Em um processo pontual, a visão estatística é de analisar a estrutura geométrica de padrões formados por objetos que são distribuídos estocasticamente no espaço (RIPLEY, 1981).

Estatisticamente, processos pontuais são definidos como um conjunto de pontos irregularmente distribuído no terreno cuja localização foi gerada por mecanismo estocástico (DIGGLE, 1983).

Ripley (1976), Diggle (1983) e Cressie (1991) descrevem detalhes relativos à teoria e aplicação de processos pontuais espaciais para diversos campos como, por exemplo, silvicultura, astronomia e biologia celular. De uma maneira geral, o objetivo inicial no processo de análise de padrão de pontos espaciais consiste em verificar se um conjunto de pontos ou eventos exhibe algum padrão sistemático, tipo regularidade, agrupamento ou aleatoriedade (RIPLEY, 1976), conforme pode ser visto na Figura 1.

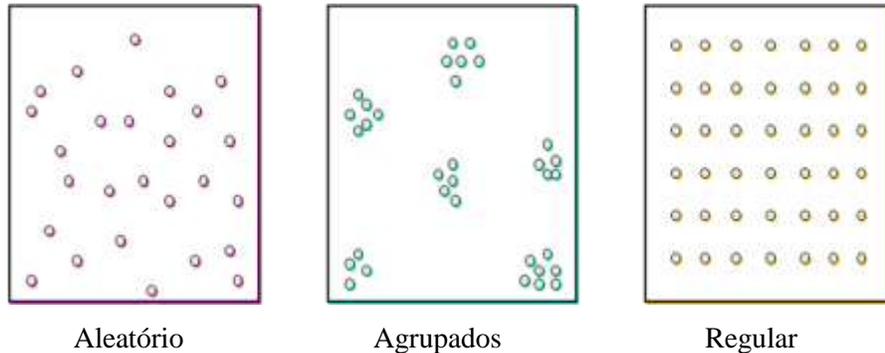


Figura 1 Três padrões de pontos simulados mostrando realização com completa aleatoriedade espacial (esquerda), agrupamentos (centro) e regularidade (direita)

O primeiro pesquisador a fazer um estudo de padrões de pontos foi o médico Britânico John Snow que, em 1845, relacionou os casos de morte por cólera em Londres. Neste estudo, Dr. Snow simplesmente localizou cada um dos casos de morte e construiu um mapa conforme apresentado na Figura 2. Pode-se observar que existe um agrupamento de casos ao redor da bomba localizada em Broady Street. Com esse resultado em mãos, ele convenceu as autoridades a interditar a bomba. Em pouco tempo, os casos de cólera em Londres caíram

drasticamente. Com esse método espacial simples, o Dr. Snow descobriu que a cólera era transmitida por água contaminada.

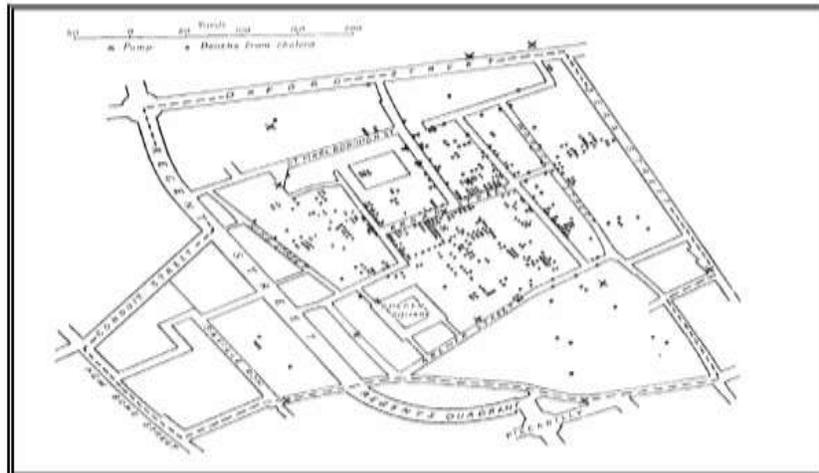


Figura 2 Mapa do Dr. John Snow mostrando os casos de óbitos por cólera, identificados por pontos e poços de água, representado por cruzes, na epidemia de cólera, em Londres, no ano de 1854

Scalon (1999) mostrou que os dados coletados por John Snow podem ser analisados sobre o moderno enfoque de análise de configurações de pontos baseado nas funções F , G e K . Com esse enfoque, pode-se determinar, inclusive, o raio de ação dentro do qual a bomba de água de Broad Street (e outras bombas) era responsável pelos casos de cólera em Londres.

2.3 Completa aleatoriedade espacial (CSR)

Diggle (2003) afirma que a análise estatística dos padrões de distribuição de pontos requer um modelo teórico de referência; base para o

desenvolvimento de métodos formais que verificam a significância dos resultados exploratórios. O modelo teórico mais simples é conhecido como completa aleatoriedade espacial (ou *Complete Spatial Randomness – CSR*), que corresponde a uma situação onde é igualmente provável que um evento aconteça em qualquer local dentro da área de estudo, embora os locais sejam de outros eventos. Quer dizer que eventos seguem uma distribuição uniforme sobre a área de estudo e são independentes um do outro. Usa-se o termo uniforme no sentido de seguir uma distribuição de probabilidade uniforme pela área de estudo, não no sentido “uniformemente” disperso na área (WALLE; GOTWAY, 2004). No modelo CSR, consideramos que as ocorrências em cada subárea são não-correlacionadas e homogêneas e estão associados à mesma distribuição de probabilidade de Poisson.

Lima (2005) afirma que para explicar a associação da distribuição de Poisson com a ocorrência aleatória de pontos em uma determinada área, pode-se considerar um raciocínio semelhante ao utilizado por Meyer (1983) para a obtenção da distribuição de Poisson a partir da distribuição binomial. Neste caso, considera-se $n \rightarrow \infty$ e $p \rightarrow 0$ de modo que $np \rightarrow \lambda$ seja constante.

Considere uma determinada área A , em alguma unidade de área conveniente, onde existem x pontos localizados aleatoriamente e, portanto, independentes. Assim, tem-se uma intensidade de $\lambda = \frac{x}{A}$ pontos por unidade de área. Suponha que a área A seja composta por n sub-regiões extremamente pequenas, de sorte que a probabilidade de que cada uma dessas sub-regiões contenha mais de um ponto é desprezível. Suponha ainda que cada um desses x pontos tenha a mesma probabilidade p de estar localizada em qualquer uma das n sub-regiões. Portanto, pode-se assumir que há x sucessos obtidos em n realizações de um ensaio, sendo X a variável aleatória que representa esse número de sucessos, $X \rightarrow Bin(n, p)$, ou seja,

$$p(X = x) = \binom{n}{x} p^x (1-p)^{n-x} I_{0,1,2,\dots,n}(x) \quad (1)$$

Sendo $\lambda = np$, que representa o número esperado de sucessos por unidade de espaço, ou seja, a intensidade ou o número de pontos por unidade de área constante e substituindo $p = \frac{\lambda}{n}$ na equação abaixo, tem-se:

$$\begin{aligned} p(X = x) &= \binom{n}{x} \left(\frac{\lambda}{n}\right)^x \left(1 - \frac{\lambda}{n}\right)^{n-x} = \frac{n!}{x!(n-x)!} \frac{\lambda^x}{n^x} \left(1 - \frac{\lambda}{n}\right)^n \left(1 - \frac{\lambda}{n}\right)^{-x} \\ &= \frac{\lambda^x}{x!} \frac{(n-1) \cdot (n-2) \cdot \dots \cdot (n-x+1)}{n^{x-1}} \left(1 - \frac{\lambda}{n}\right)^n \left(1 - \frac{\lambda}{n}\right)^{-x}. \end{aligned}$$

Note que,

$$\begin{aligned} \frac{(n-1) \cdot (n-2) \cdot \dots \cdot (n-x+1)}{n^{x-1}} &= \frac{(n-1)}{n} \cdot \frac{(n-2)}{n} \cdot \dots \cdot \frac{(n-x+1)}{n} \\ &= \left(1 - \frac{1}{n}\right) \cdot \left(1 - \frac{2}{n}\right) \cdot \dots \cdot \left(1 - \frac{x-1}{n}\right) \end{aligned}$$

e, portanto,

$$P(X = x) = \frac{\lambda^x}{x!} \left(1 - \frac{1}{n}\right) \cdot \left(1 - \frac{2}{n}\right) \cdot \dots \cdot \left(1 - \frac{x-1}{n}\right) \left(1 - \frac{\lambda}{n}\right)^n \left(1 - \frac{\lambda}{n}\right)^{-x}$$

Considerando $n \rightarrow \infty$, o que estaria relacionado a um número grande de sub-

regiões, o termo $\left(1 - \frac{\lambda}{n}\right)^n \rightarrow e^{-\lambda}$, pela própria definição do número e e os

termos $\left(1 - \frac{a}{n}\right) \rightarrow 1$, em que $a = 1, 2, \dots, x-1$ e $a = \lambda$. Assim,

$$P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!} I_{\{0,1,2,\dots\}}(x), \quad (2)$$

que corresponde à distribuição de Poisson.

Numa visão intuitiva, pode-se considerar que a posição dos eventos é independente e que os eventos têm igual probabilidade de ocorrência em toda a região A .

2.4 Processos Pontuais com agrupamentos

No mundo natural, padrões agrupados são muito comuns, talvez mais comuns do que padrões regulares ou aleatórios. Por exemplo, algumas árvores jovens em florestas naturais e galáxias no universo tendem a formar agrupamentos. Em algumas áreas, a densidade de pontos é muito alta, isto é, os agrupamentos de pontos são rodeados por áreas de baixa densidade de pontos, talvez até mesmo por espaços vazios. A definição de um agrupamento é um pouco subjetiva: agrupamentos são grupos de pontos com uma distância entre pontos inferior à distância padrão comum. Esta agregação local simplesmente não pode ser resultado de flutuações aleatórias dos pontos (STOYAN, 2008).

Conforme afirma Stoyan (2008), em geral, agrupamentos podem ser causados por diferentes processos:

1. os pontos se espalharam originalmente em algumas sub-regiões, dentro da região de interesse, que são distribuídas irregularmente no espaço. Plantas cujas sementes só germinam em áreas onde o ambiente oferece condições satisfatórias são um exemplo clássico disso.

2. o padrão de agrupamento é resultado de um processo físico. Leis físicas determinam a formação de agrupamentos. Um exemplo é o caso de galáxias.

3. o padrão é um resultado de um mecanismo estocástico que envolve pais e filhos, sendo que os filhos se localizam ao redor dos pais.

4. o padrão é resultado da interação positiva entre os pontos. Com o passar do tempo, a densidade de pontos aumenta em áreas onde a densidade era inicialmente mais alta e a densidade diminui em áreas onde a densidade era inicialmente mais baixa. Cooperação entre plantas pode ser um exemplo disto.

O algoritmo de criação de processos pontuais com agrupamento introduzido por Neyman e Scott (1958) incorpora uma forma explícita de agrupamento espacial. A definição incorpora três postulados:

P1 - Cada evento pai forma um processo de Poisson com intensidade ρ

P2 - Cada pai produz um número aleatório S de descendentes, independentes e idênticos para cada pai de acordo com uma distribuição de probabilidade p_s :
 $s=0,1,\dots$

P3 - Em relação ao pai, a posição dos descendentes é independente e identicamente distribuída, de acordo com uma função densidade de probabilidade.

Quando P2 especificar uma distribuição de Poisson para o número de descendentes por pai, se for preciso, o processo pode ser simulado condicionado ao número total de eventos dentro de uma área A , alocando os eventos aleatoriamente entre os pais.

A figura 3 apresenta uma realização de uma configuração com agrupamentos em área unitária com 10 grupos (pais) com 12 eventos cada (descendentes) e uma variabilidade radial de 0,02 por grupo

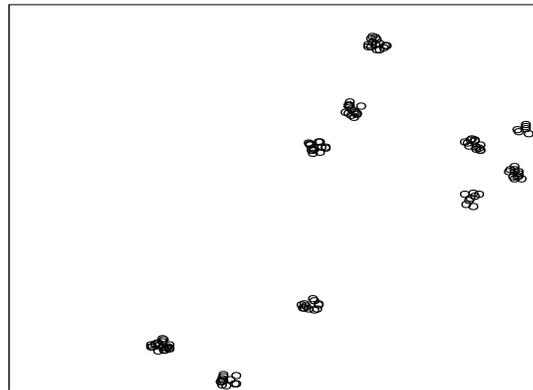


Figura 3 Realização de um processo pontual em uma área unitária com 10 grupos, cada grupo com 12 eventos e variabilidade radial de 0.02

2.5 Processos Pontuais com regularidade

Processos pontuais exibindo regularidade são úteis para modelar situações em que os eventos são forçados por algum fenômeno a apresentar, pelo menos, certa distância mínima entre eles. Em especial, esses processos são úteis para modelar a localização dos centros de tamanho finito de partículas circulares que não podem ficar sobrepostas.

Para gerar uma realização de pontos com regularidade, dentro de uma região, Diggle (2003) sugere o algoritmo denominado de *Simple Sequential Inhibition* (SSI). Começando com uma região vazia, o algoritmo adiciona pontos, um por um. Cada novo ponto é gerado uniformemente na região, independentemente dos pontos anteriores. Se o novo ponto gerado está mais

próximo (distância menor que r) de um ponto já existente, então ele é rejeitado e um novo ponto é gerado aleatoriamente. O algoritmo termina quando o número desejado de pontos é alcançado, ou quando não é possível incluir qualquer ponto distante dos demais com uma distância superior a r .

A Figura 4 mostra uma realização de uma configuração apresentando regularidade, com distância mínima de 0,05 entre os pontos em uma área unitária.

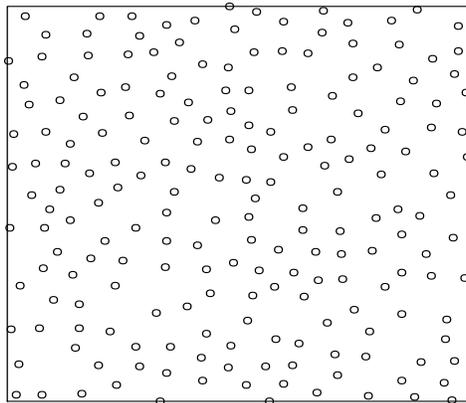


Figura 4 Realização de uma configuração com regularidade em uma área unitária com 200 pontos e distância mínima entre eles de $r = 0,05$

2.6 Descrição de processos pontuais espaciais

Um processo pontual pode ser descrito em termos dos efeitos de primeira e segunda ordem (DIGGLE, 2003). Segundo Câmara et al. (2002), os efeitos de primeira ordem, considerados globais ou de larga escala, correspondem a variações no valor médio do processo pontual no espaço. Neste caso, estamos interessados na intensidade do processo, isto é, no número de eventos por unidade de área.

Segundo Bayley e Gatrell (1995), os efeitos de segunda ordem -- denominados locais ou de pequena escala -- representam a dependência espacial do processo, proveniente da estrutura de correlação espacial. Para medir a dependência espacial, procuramos estimar o relacionamento entre pares de eventos (por unidade de área) no espaço, o que corresponde a uma aproximação do cálculo da covariância entre as variáveis aleatórias que representam cada evento.

Considerem um conjunto de pontos (u_1, u_2, \dots) numa determinada região A . O processo pontual é modelado considerando-se sub-regiões S em A , através de sua esperança $E N(S)$ e a covariância $C[N(S_i), N(S_j)]$, onde $N(S)$ denota o número de eventos em S . Uma vez que o objetivo da análise é estimar as localizações prováveis de ocorrência de determinados eventos, essas estatísticas devem ser inferidas considerando o valor limite da quantidade de eventos por área. Este valor limite corresponde à esperança de $N(S)$ para uma pequena região du em torno do ponto u , quando essa tende a zero. Essa esperança é denominada intensidade (propriedade de primeira ordem), sendo definida como:

$$\lambda(u) = \lim_{|du| \rightarrow 0} \left\{ \frac{E N(du)}{|du|} \right\}. \quad (3)$$

Propriedades de segunda ordem podem ser definidas da mesma forma, considerando a intensidade conjunta $\lambda = (u_i, u_j)$ entre duas regiões infinitesimais $|du_i|$ e $|du_j|$ que contêm os pontos u_i e u_j .

$$\lambda(d(u_i), d(u_j)) = \lim_{du_i, du_j \rightarrow} \left\{ \frac{C[N(d(u_i), N(d(u_j)))]}{du_i, du_j} \right\}. \quad (4)$$

Quando o processo é estacionário, $\lambda(u)$ é uma constante, isto é $\lambda(u) = \lambda$; se também é isotrópico, $\lambda = (u_i, u_j)$ se reduz a $\lambda(|h|)$, sendo que $|h|$ é a distância entre os dois pontos. Quando o processo é não estacionário, ou seja, a intensidade média varia na região A , a modelagem da estrutura de dependência $\lambda = (u_i, u_j)$ incorpora a variação de $\lambda(u)$. De forma explícita ou implícita, a maior parte das técnicas de análise de distribuição de pontos supõe um comportamento estacionário e isotrópico do processo aleatório subjacente aos eventos analisados.

Para um processo estacionário e isotrópico existe muito interesse em considerar funções de distribuições que caracterizam o processo pontual espacial, As funções mais importantes são F , G , J e K .

2.6.1 Função F

Introduzida por Ripley (1977), a função F é a função de distribuição acumulada da distância entre um ponto aleatoriamente escolhido (não o evento) e o evento mais próximo. Mesmo não corrigindo o efeito da borda, o mais simples estimador para a função F pode ser obtido pela equação

$$\hat{F}(x) = \lambda^{-1} \sum_{i=1}^m I_x(x_i), \quad (5)$$

em que m é o número de pontos arbitrários no mapa, x_i representa a distância do i -ésimo ponto aleatório para o mais próximo dos n eventos do mapa

analisado e $I_x(x_i)$, uma função indicador igual a 1 quando x_i é menor ou igual a x e 0 caso contrário.

2.6.2 Função G

Introduzida por Ripley (1977) e conhecida na literatura como o método do vizinho mais próximo, a função G é a função de distribuição acumulada entre a distância de um evento e o seu vizinho mais próximo. O mais simples estimador para a função G pode ser obtido pela equação

$$\hat{G}(y) = \lambda^{-1} \sum_{i=1}^n I_y(y_i), \quad (6)$$

em que n é o número de eventos no mapa, y_i representa a distância do i -ésimo evento para o mais próximo dos n eventos do mapa analisado e $I_y(y_i)$ uma função indicador igual a 1 quando y_i é menor ou igual a y e 0 caso contrário.

2.6.3 Função J

Proposta por Van Lieshout e Baddeley (1996), a função J é a razão da função de distribuição acumulada entre a distância de um evento e o vizinho mais próximo (função G) e a função de distribuição acumulada da distância entre um ponto aleatoriamente escolhido e o evento mais próximo (função F), que compara as distâncias entre os eventos por meio de distâncias fixadas (z) de um ponto amostral. A função J é definida por:

$$J(z) = \frac{1-G(z)}{1-F(z)}, \quad (7)$$

para todo $z > 0$ e $F(z) < 1$.

2.6.4 Função K de Ripley

Proposta por Ripley (1977), a função K é uma estatística descritiva do padrão espacial, também chamada de “análise de segunda ordem reduzida”, pois está relacionada com as propriedades de segunda ordem.

Para o processo univariado, a função K é definida como: $\lambda K(r) = E(\text{número de eventos contidos a uma distância } r \text{ de um evento arbitrário})$, em que $E(\cdot)$ é a esperança, e λ é a intensidade, ou número médio de eventos por unidade de área, assumida constante na região (RIPLEY, 1976).

Um estimador de $K(r)$, que corrija o efeito de bordas, é dado por

$$\hat{K}(r) = \frac{A}{n^2} \sum_i^n \sum_{j, i \neq j}^n \frac{I_r(d_{(ij)})}{w_{ij}} \quad (8)$$

em que A é a área da região, n é o número de eventos observados, $d_{(i,j)}$ é a distância entre os evento i e j , $I_r(d_{(i,j)})$ é uma função indicadora cujo valor é 1 se $I_r(d_{(i,j)}) \leq r$ e 0 em caso contrário, e w_{ij} é a proporção da circunferência do círculo centrado no evento i que está dentro da região (correção devido ao efeito de borda).

A função K de Ripley computa todas as distâncias $d_{(i,j)}$ entre todos os pontos de um mapa para investigar seu padrão. Para entender melhor essa função, observe a Figura 5. Imagina-se que cada evento (ponto) é visitado e, ao seu redor, constroem-se círculos concêntricos espaçados. O número acumulado

de eventos dentro de cada um desses círculos é contado. Todos os eventos são visitados e o número de eventos que caem dentro de uma distância r , dentre todos os eventos, é calculado, passando a ser a estimativa da função K , quando ponderada por $\frac{A}{n^2}$.

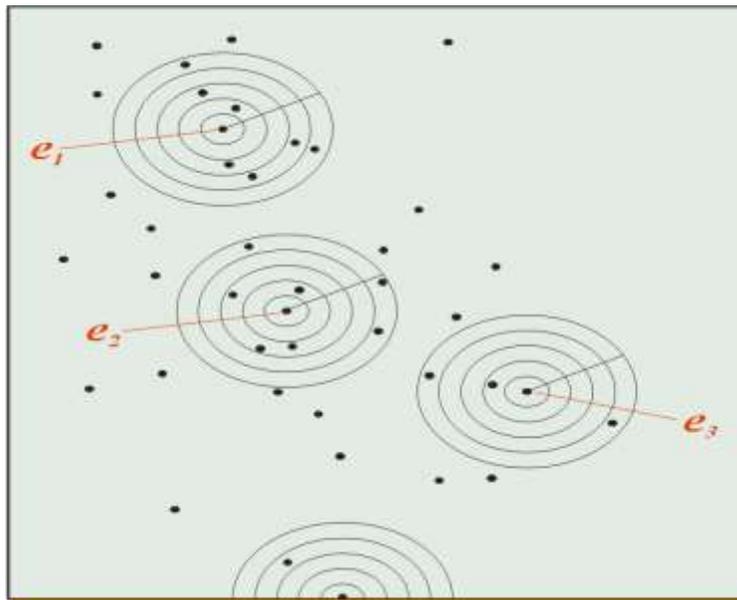


Figura 5 Ideia gráfica do estimador da Função K

Pode-se observar que a função K é definida como o número esperado de eventos dentro de uma distância r de um evento arbitrário dividido pela intensidade de pontos na região. A sua estimação baseia-se na variação do número médio de outros eventos em torno de um evento qualquer, à medida que variamos o raio de consideração. Sob a hipótese de Completamente Aleatória Espacial (CSR), a função K é expressa como $K(r) = \pi r^2$.

A função K é uma das ferramentas da estatística espacial mais apropriadas para a análise de padrão de pontos quando os dados são apresentados sob a forma de mapas (a posição de todos os eventos dentro da área é conhecida).

Entre as principais vantagens da função K estão as que permitem a detecção do padrão espacial em diferentes escalas de distâncias simultaneamente e que o padrão espacial observado seja comparado aos modelos conhecidos, possibilitando, assim, investigar o nível de independência espacial entre os grupos de pontos quaisquer.

Se compararmos a função K a outros métodos frequentemente utilizados no processo pontual, comprovamos seu potencial de uso e sua robustez estatística. Métodos baseados em áreas ou distâncias dos vizinhos mais próximos, que são mais comuns, carregam em sua essência “simplificações” da realidade, uma vez que tratam apenas da simples contagem de indivíduos dentro de parcelas ou pela procura pelo primeiro vizinho mais próximo (RIPLEY, 1976). Desse modo, muitas informações são perdidas nas análises espaciais uma vez que tais métodos só conseguem detectar o padrão espacial naquela escala particular em que os dados foram coletados, como o tamanho da parcela ou a distância entre os pontos.

2.6.5 Mosaico de Dirichlet

Um mosaico de Dirichlet é a divisão de uma região do plano em pequenas sub-regiões não sobrepostas que são chamadas de *tiles*. Um mosaico de Dirichlet lembra uma parede coberta com azulejos de diferentes formas.

De uma maneira mais formal, seja X_1, \dots, X_n uma configuração pontual no plano cartesiano. O mosaico de Dirichlet divide a área de amostragem da configuração pontual em regiões (polígonos), de tal forma que a

região D_i contém apenas o ponto X_i . Além disso, todos os polígonos vizinhos de D_i estão mais próximos de X_i do que qualquer outro ponto dentro da área de amostragem. Este mosaico foi proposto em 1850 por Dirichlet (UPTON; FINGLETON, 1985) e uma definição matemática formal é dada por Green e Sibson (1978).

O mosaico de Dirichlet -- também conhecido como Voronoi -- foi reinventado várias vezes e é útil para a investigação em muitos domínios científicos. Upton e Fingleton (1985) afirmam que o mosaico de Dirichlet determinado por um retículo irregular é uma das mais úteis construções. Uma representação gráfica do mosaico de Dirichlet está apresentada na Figura 6.

Os mosaicos de Dirichlet podem ser pensados como áreas territoriais de animais ou um conjunto de células em um tecido. Por exemplo, dois pontos adjacentes, representando animais competitivos, de igual força, podem dividir a área plana entre eles, bem como com quaisquer outros animais próximos. Em geral, os concorrentes que estão mais longe de um organismo serão menos suscetíveis à interferência no espaço, a menos que não existam outros organismos competindo entre as áreas. Assim, os mosaicos de Dirichlet, geralmente, devem coincidir com as áreas territoriais (KENKEL; HOSKINS; HOSKINS, 1989).

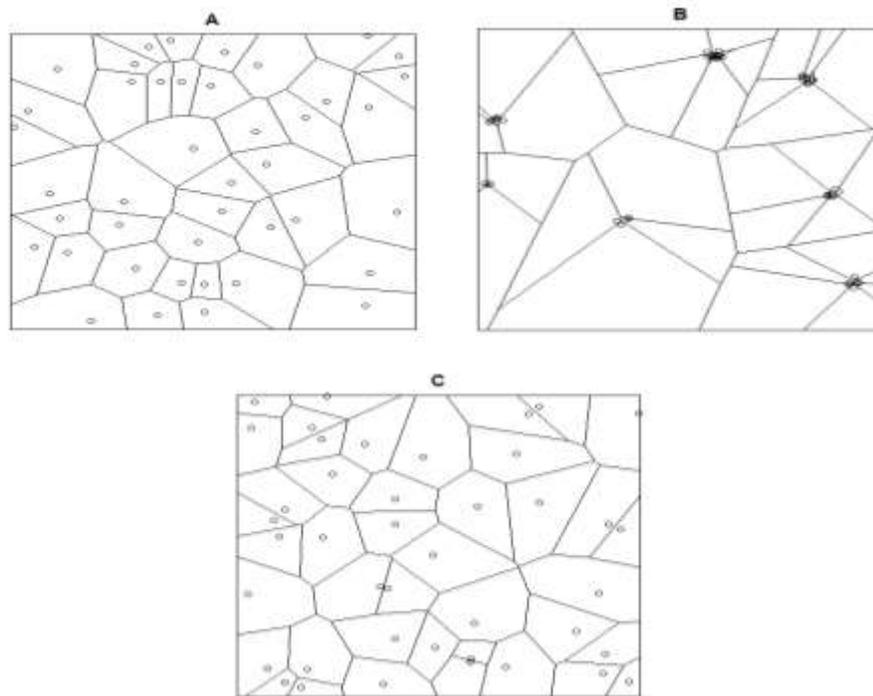


Figura 6 Mosaicos de Dirichlet para 42 pontos. Sendo regularidade (A), agrupamento(B) e aleatoriedade (C)

Pode-se observar que os tamanhos das áreas dos mosaicos fornecem uma ideia do tipo de configuração espacial. Se as áreas forem aproximadamente iguais, temos evidências de regularidade. Caso algumas regiões do plano apresentem mosaicos com áreas muito pequenas, então existem indícios de agrupamentos nessas regiões.

O primeiro algoritmo computacional para desenhar os mosaicos de Dirichlet foi sugerido por Green e Sibson (1978). Estes autores desenvolveram um programa em ANSI Fortran para uso em computadores *mainframe* que foi, posteriormente, utilizado em livros didáticos de estatística espacial (DIGGLE, 2003; RIPLEY, 1981).

Atualmente, vários softwares como o SYSTAT (WILKINSON, 1990) e o R (R DEVELOPMENT CORE TEAM, 2010) oferecem facilidades para construir, desenhar e calcular as áreas dos mosaicos de Dirichlet.

2.7 Testes de hipóteses e poder

Nesta dissertação, pretende-se construir testes de hipóteses para a completa aleatoriedade espacial utilizando funções de distribuições que caracterizam o processo pontual espacial, mais especificamente as funções K e H . Além disso, pretende-se avaliar o poder desses testes. Assim, neste item, faz-se uma introdução sobre testes de hipóteses e o poder desses testes sob o ponto de vista da estatística clássica.

Seja H_0 a hipótese nula existente a ser testada (que, neste trabalho, representa a completa aleatoriedade espacial) e H_1 a hipótese alternativa (representada neste trabalho por configurações que exibem agrupamentos e regularidade). Portanto, H_1 é a hipótese complementar de H_0 . O teste concluirá pela aceitação ou rejeição da hipótese nula, implicando a negação ou a afirmação de H_1 .

Ao testar a hipótese de nulidade, o pesquisador corre o risco global de tomar decisões erradas e tal risco deverá ser considerado. Ao decidir se rejeita ou não a hipótese de nulidade, o pesquisador está sujeito a incorrer em erros de decisão que são os erros do tipo I e II (MOOD et al., 1974).

O erro tipo I é o erro cometido ao se rejeitar a hipótese nula verdadeira, quando esta deveria ser aceita e é diretamente controlável pelo pesquisador. A probabilidade de se cometer esse erro é dada por $\alpha = P(\text{erro tipo I}) = P(\text{rejeitar } H_0, \text{ quando } H_0 \text{ é verdadeira})$. O erro tipo I é, em geral, denominado nível de significância.

O erro tipo II é o erro cometido ao não se rejeitar a hipótese nula falsa. Não é controlável diretamente pelo pesquisador. A probabilidade de se cometer esse tipo de erro é dada por $\beta = P(\text{erro tipo II}) = P(\text{não rejeitar } H_0 \text{ quando } H_0 \text{ é de fato falsa})$.

O poder de um teste é definido como a probabilidade de aceitar a hipótese alternativa quando ela é verdadeira, isto é, $1 - \beta$ e este valor depende dos parâmetros que estão sendo testados.

Observa-se que as probabilidades de se cometerem os erros tipo I e II são inversamente proporcionais, sendo impossível controlá-las ao mesmo tempo em um único experimento. Por isso, cabe ao pesquisador avaliar os objetivos propostos para fazer a sua opção. Portanto, é necessário manter certo equilíbrio, pois, ao controlar de maneira excessiva a taxa do erro tipo I, a taxa do erro tipo II aumenta e o poder do teste diminui, levando o pesquisador a aceitar como iguais os efeitos dos processos pontuais. Quando isso ocorre, o teste é considerado conservativo (CARMER; SWANSON, 1973). Mais se acontecer um aumento significativo da taxa do erro tipo I (maior que α), então a taxa do erro tipo II diminui e o poder do teste aumenta. Portanto, o teste é considerado poderoso e liberal (HOCHBERG; TAMHANE, 1987).

Teoricamente, algumas medidas podem ser tomadas para minimizar a taxa do erro tipo II, como a escolha apropriada do teste e da avaliação das suas pressuposições, que devem ser atendidas. Se essas condições forem satisfeitas, existe uma garantia de maior poder. Outra medida que pode tornar o teste mais poderoso é a determinação do tamanho da amostra, sem aumentar demasiadamente o custo da pesquisa. Por último, pode-se fixar o nível de significância α entre 0,10 e 0,01, devido à relação inversa entre as taxas do erro tipo I e erro tipo II (FERREIRA, 2009).

O poder de um teste de hipóteses é afetado por dois fatores: i) tamanho da amostra, isto é, mantendo-se todos os outros parâmetros iguais, quanto maior

o tamanho da amostra, maior o poder do teste; ii) nível de significância - quanto maior o nível de significância, maior o poder do teste. Ao elevar o nível de significância, reduz-se a região de aceitação. Como resultado, crescem as chances de rejeição da hipótese nula. Isto significa há menos chances de aceitação da hipótese nula quando ela é falsa, isto é, menor chance de cometer um erro do tipo II, aumentando o poder do teste. O verdadeiro valor do parâmetro a ser testado: Quanto maior a diferença entre o “verdadeiro” valor do parâmetro e o valor especificado pela hipótese nula, maior o poder.

As características probabilísticas de um teste podem ser descritas através de uma função que associa a cada valor do parâmetro θ a probabilidade $\pi(\theta)$ de rejeitar H_0 . A função $\pi(\theta)$ é chamada função de poder (ou potência) do teste. Assim, denotando por C a região crítica da função de poder é definida como:

$$\pi(\theta) = P(X \in C / \theta), \forall \theta \in \Theta \quad (9)$$

A função poder é a ferramenta utilizada para verificar a adequação de um teste ou para comparar dois ou mais testes. Evidentemente, uma função de poder ideal seria tal que $\pi(\theta) = 0$ para θ satisfazendo H_0 e $\pi(\theta) = 1$ para θ satisfazendo H_1 . Em um problema prático, no entanto, raramente existirá um teste com estas características.

Para efeito de ilustração, a função poder está representada na Figura 7 para o exemplo: $P(X \leq 6 / p)$ para $0 < p < 1$, onde $X \sim \text{Binomial}(10, p)$. Note que, neste exemplo, se p for maior do que, por exemplo, 0,8, então o teste aceitará H_0 , indicando que o teste é adequado. Por outro lado, para valores de p entre 0,7 e 0,8, o teste ainda rejeita H_0 com probabilidade baixa. Foi usado um exemplo da Binomial por sua aproximação com a Poisson, que

está diretamente relacionada à hipótese de completa aleatoriedade espacial, conforme apresentada no item 2.3.

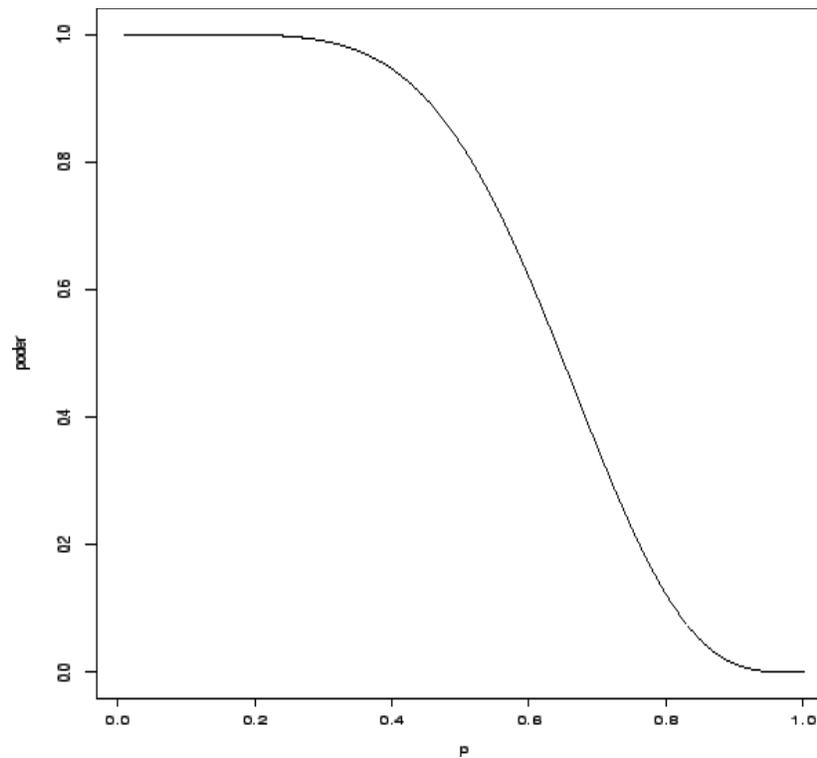


Figura 7 Função de poder da distribuição binomial

2.8 Análise da completa aleatoriedade espacial usando a função K

A análise da completa aleatoriedade espacial (CSR) é o primeiro passo na análise de padrões de pontos. Conforme Diggle (2003), se a hipótese de aleatoriedade espacial for aceita, tem-se duas consequências principais: (1) não há necessidade de se considerar um modelo mais complicado que o modelo de Poisson. (2) não é possível encontrar indicadores de interação interessantes entre os pontos com baseando-se apenas na geometria do padrão observado se

não houver qualquer informação ou dados adicionais sobre os processos subjacentes.

Um grande número de métodos para avaliar a aleatoriedade foi desenvolvido e a pesquisa nesta área ainda está em curso. Todas as funções descritivas dos processos pontuais no espaço (F, G, K, J) podem ser usadas para construir esses testes. A experiência mostra que não é possível obter um melhor ‘teste’ usando apenas uma das funções. Em geral, usa-se a combinação de testes baseada em, pelo menos, duas funções (DIGGLE, 2002).

Neste item, apresentam-se métodos para testar a hipótese nula de completa aleatoriedade espacial de uma configuração espacial de pontos, usando a função K . Os mesmos métodos podem ser desenvolvidos, de maneira análoga, para as outras funções. Esses métodos estão descritos em detalhes em Diggle (2002).

Sabe-se que, para um processo aleatório, a função K é dada por $K(r) = \pi r^2$ enquanto um estimador da função K é dado pela equação (8). Portanto, a comparação entre as duas funções é uma forma de analisar a completa aleatoriedade espacial. Para melhorar a interpretação, é usual utilizar um estimador linear da função K , denominada função L e definida pela equação (10).

$$\hat{L}(r) = \sqrt{\frac{\hat{K}(r)}{\pi}} \quad (10)$$

O estimador definido em (10) pode ser utilizado de duas formas (gráfico e teste formal) para testar a hipótese nula de que os eventos seguem uma distribuição de Poisson homogênea, ou seja, que os eventos estão aleatoriamente distribuídos no plano. Segundo Scalón et al. (2003) e Diggle (2002), estes procedimentos podem ser feitos utilizando-se a simulação Monte Carlo.

Para a construção exata do teste contra a hipótese de completa aleatoriedade espacial, primeiramente, defina u_i como a medida da discrepância entre $\hat{L}_i(r)$ e r sobre toda distância. Esta medida pode ser obtida tanto pela distância de Cramer-VonMises, como pela distância de Kolmogorov dadas, respectivamente, por

$$L_M = \int_0^{r_0} \hat{L}_i(r) - r^2 dr \quad (11)$$

$$L_I = \max_{0 \leq r \leq r_0} |\hat{L}_i(r) - r|, \quad (12)$$

para $u_i : i = 1, 2, \dots, s$, onde u_0 é a medida para o processo pontual observado. Se, ao ordenar u_i obtivermos u_0 em um dos extremos da distribuição, existem evidências estatísticas para rejeitar a hipótese de completa aleatoriedade espacial.

O teste apresentado anteriormente não fornece a direção (regularidade ou agrupamento) da rejeição da hipótese nula. O procedimento gráfico apresentado a seguir resolve esse problema.

Seja $\hat{L}_i(r)$ calculada a partir dos dados de uma configuração pontual observada. Calcule a função de distribuição empírica $\hat{L}_i(t) : i = 2, 3, \dots, s$, para cada $(s-1)$ simulações independentes de n eventos independentes e identicamente distribuídos na região observada sob a suposição de completa aleatoriedade espacial e defina os envelopes de simulação superiores e inferiores como:

$$U(r) = \max \hat{L}_i(r), i = 1, \dots, s \quad (13)$$

$$L(r) = \min \hat{L}_i(r), i = 1, \dots, s \quad (14)$$

Estes envelopes de simulações são colocados em um gráfico juntamente com o gráfico de $\hat{L}_1(r)$, seguindo a propriedade

$$P \hat{L}_1(r) > U(r) = P \hat{L}_1(r) < L(r) = s^{-1} \quad (15)$$

Os valores de $\hat{L}_1(r)$, $U(r)$ e $L(r)$ são colocados no eixo y do gráfico, enquanto as distâncias r são colocadas no eixo x . Se a linha de $\hat{L}_1(r)$ estiver acima do envelope superior, evidencia a ocorrência de agrupamento na escala considerada. Caso $\hat{L}_1(r)$ fique abaixo do envelope inferior, há evidências estatísticas de regularidade nessa escala.

2.9 Análise da completa aleatoriedade espacial usando mosaicos de Dirichlet

A função K representa um instrumento valioso e natural para a descrição de um processo pontual espacial, mas não fornece uma descrição completa da estrutura espacial. É possível encontrar processos diferentes que têm funções K idênticas (CRESSIE, 1991). Para uma descrição completa de um padrão espacial pontual seria necessário incluir outros descritores, como as funções F , G e J .

Essa dissertação apresenta uma alternativa às funções clássicas F , G , K e J para testar a completa aleatoriedade espacial. Esta alternativa tem como base as áreas do mosaico de Dirichlet. Myles et al. (1995) foram os primeiros a propor os mosaicos de Dirichlet como ferramenta para análise de configurações espaciais de pontos.

Conforme foi visto no item 2.6.5, um mosaico de Dirichlet circunda um ponto como um polígono plano em que todas as regiões vizinhas estão mais perto de um ponto do que para outros pontos. Claramente, os tamanhos e as formas dos polígonos dependem do arranjo dos eventos. Tamanhos e formas quase iguais são indícios de configurações regulares, enquanto tamanhos muito pequenos e diferentes podem fornecer indícios de uma configuração com agrupamentos.

O problema é determinar estatisticamente o quanto os tamanhos e as formas dos polígonos podem ser considerados iguais (ou diferentes) para rejeitar a hipótese nula de uma configuração gerada por um processo espacial completamente aleatório. Para tal, inicialmente, calcula-se a área (U) de cada sub-região do mosaico de Dirichlet. Então, estima-se $H(r)$, ou seja, a função distribuição das subáreas do mosaico de Dirichlet dado por

$$\hat{H}(r) = \frac{U(r)}{n}, \quad (16)$$

em que $U(r)$ é o número total de mosaicos com áreas inferiores a r .

A título de ilustração, apresenta-se um exemplo do cálculo de $\hat{H}(r)$ para um mosaico de 15 pontos, conforme Figura 8.

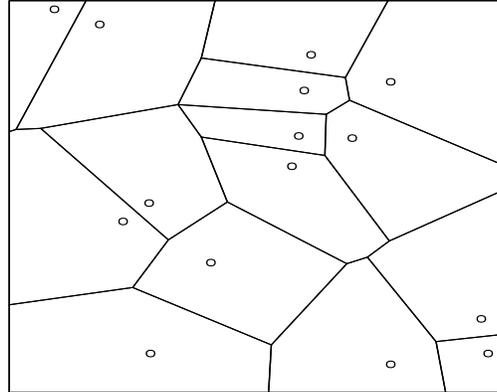


Figura 8 Mosaico de Dirichlet com 15 pontos aleatórios

Supõe-se que as áreas dos polígonos sejam as seguintes: 1, 1, 2, 3, 3, 4, 4, 5, 5, 5, 6, 7, 7, 8 e 8. Utilizando a equação (15), obtém-se os seguintes resultados:

$$\hat{H}(1) = \frac{U(1)}{15} = \frac{0}{15} = 0$$

$$\hat{H}(2) = \frac{U(2)}{15} = \frac{2}{15} = 0,13333\dots$$

$$\hat{H}(3) = \frac{H(3)}{15} = \frac{3}{15} = 0,2$$

$$\hat{H}(4) = \frac{H(4)}{15} = \frac{5}{15} = 0,33333\dots$$

$$\hat{H}(5) = \frac{U(5)}{15} = \frac{7}{15} = 0,46666\dots$$

$$\hat{H}(6) = \frac{U(6)}{15} = \frac{10}{15} = 0,6666\dots$$

$$\hat{H}(7) = \frac{U(7)}{15} = \frac{11}{15} = 0,7333\dots$$

$$\hat{H}(8) = \frac{U(8)}{15} = \frac{13}{15} = 0,9666\dots$$

$$\hat{H}(9) = \frac{U(9)}{15} = \frac{15}{15} = 1,0000$$

Um problema com o uso de $\hat{H}(r)$ para a construção de um teste contra a hipótese de aleatoriedade espacial é que o valor esperado $E[\hat{H}(r)]$ para o processo estocástico que representa a completa aleatoriedade espacial não é conhecido, como no caso das funções F , G , J e K .

Para contornar esse problema, pode-se gerar uma série de configurações sob a hipótese de completa aleatoriedade espacial e calcular $\hat{H}(r)$ para cada uma das configurações. A seguir, estima-se $E[\hat{H}(r)]$ pelo valor médio do conjunto.

Para construção exata do teste contra a hipótese de completa aleatoriedade espacial, primeiramente, defina H_i como a medida da discrepância entre $\hat{H}(r)$ e $\hat{E}[\hat{H}(r)]$ sobre todas as áreas de tamanho r . Esta medida pode ser obtida por várias medidas de distância. Este trabalho utiliza as distâncias de Cramer-VonMises e Kolmogorov.

Usando estas duas distâncias, obtém-se as estatísticas de teste

$$H_M = \max_{0 \leq r \leq r_0} \left| \hat{H}(r) - \hat{E}[\hat{H}(r)] \right|, \quad 17$$

$$H_I = \int_0^{r_0} \left(\hat{H}(r) - \hat{E}[\hat{H}(r)] \right)^2 dr \quad 18$$

Infelizmente, as equações (17) e (18) não têm distribuições amostrais conhecidas. Portanto, como no caso da função K , pode-se utilizar o seguinte procedimento Monte Carlo para testar a hipótese nula de completa aleatoriedade espacial. Para tal, considera-se H_i qualquer uma das estatísticas de teste (H_M ou H_I) e seguem-se os seguintes passos: (i) Calcular H_i para a configuração observada. (ii) Simular S independentes configurações espaciais sob hipótese

nula de completa aleatoriedade espacial com a mesma intensidade da configuração observada. (iii) Calcule H_i para cada configuração aleatória simulada ($i = 2, \dots, S$). Compare o valor H_l com os valores H_i ($i = 2, \dots, S$) para as configurações simuladas. Se H_l se coloca entre os maiores valores de H_2, H_3, \dots, H_s , então há indícios de que a hipótese nula de aleatoriedade espacial seja rejeitada. Suponha que para $H_l = H_{(j)}$ para algum $j \in \{1, \dots, s\}$, então se rejeita a hipótese de completa aleatoriedade espacial se $P_{value} = \frac{(s+1-j)}{s} \leq \alpha$, onde α é o nível de significância.

Assim como na função K , os testes propostos anteriormente não indicam a direção da rejeição da hipótese nula. Para tal, pode-se utilizar um procedimento gráfico para detectar a direção da rejeição da hipótese de aleatoriedade espacial.

Seja $\hat{H}_1(r)$ calculada a partir dos dados de uma configuração pontual observada. Calcule a função de distribuição empírica $\hat{H}_i(r) : i = 2, 3, \dots, s$ para cada $(s-1)$ simulações independentes de n eventos independentes e identicamente distribuídos na região observada sob a suposição de completa aleatoriedade espacial e defina os envelopes de simulação superiores e inferiores como:

$$U(r) = \max \hat{H}_i(r) , i = 1, \dots, s \quad (19)$$

$$L(r) = \min \hat{H}_i(r) , i = 1, \dots, s \quad (20)$$

Os valores de $\hat{H}_1(r)$, $U(r)$ e $L(r)$ são colocados no eixo y do gráfico, enquanto as áreas r são colocadas no eixo x. Se a linha de $\hat{H}_1(r)$ estiver acima do envelope superior, então a ocorrência de agrupamentos na escala de área

considerada é evidente. Caso $\hat{H}_1(r)$ fique abaixo do envelope inferior, há evidências estatísticas de regularidade nessa escala de área.

As Figuras 9 e 10 mostram os comportamento das funções K de Ripley e H de Dirichlet em configurações com regularidade e agrupamentos.

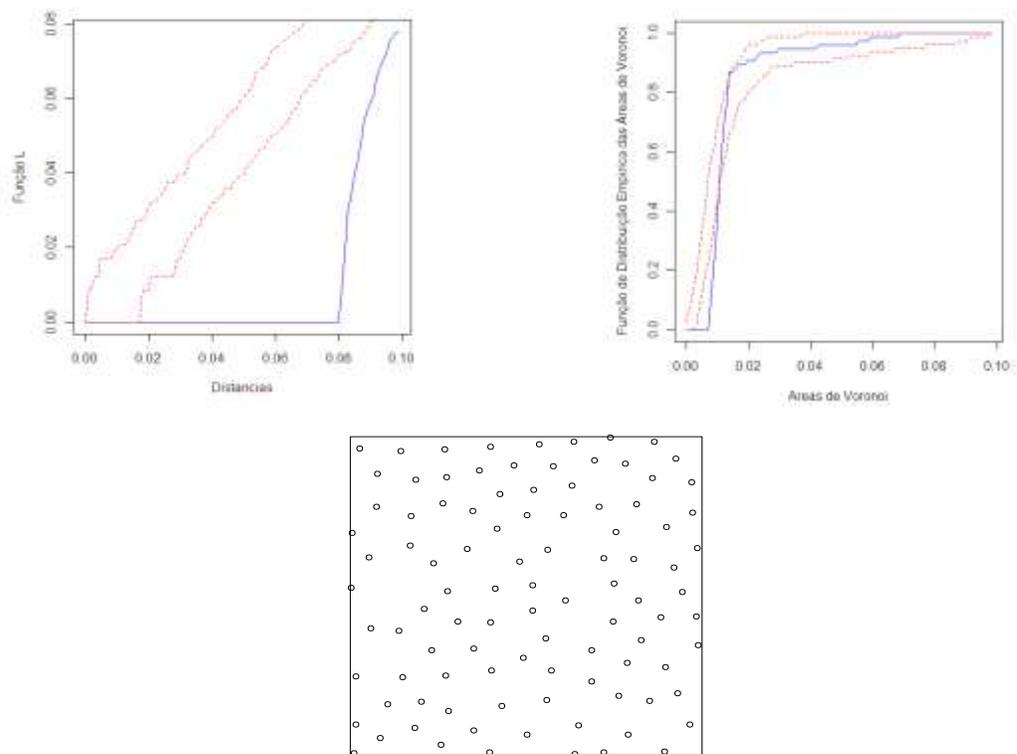


Figura 9 Simulação de um processo de regularidade com 100 pontos e distância radial de 0,08, usando as funções L de Ripley e H de Dirichlet

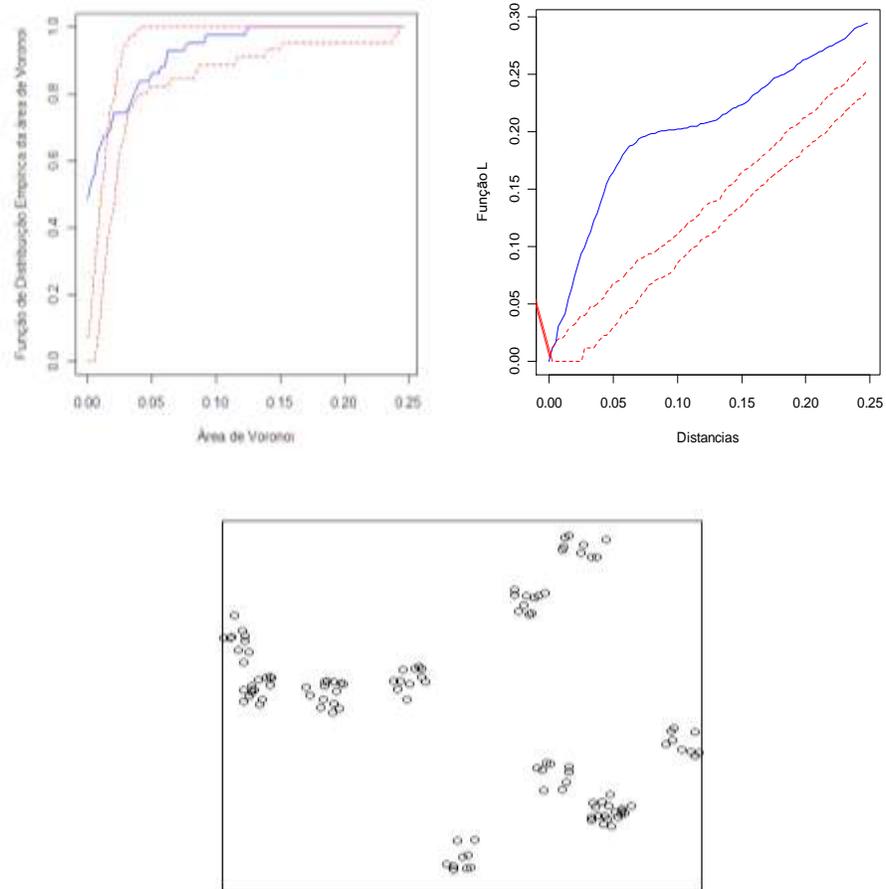


Figura 10 Simulação de um processo de agrupamento com 100 pontos, 10 grupos com 10 pontos cada, variabilidade radial 0,05, usando as funções L de Ripley e H de Dirichlet

Observa-se pelas Figuras 9 e 10 que as funções L e D conseguiram identificar corretamente o processo estocástico gerador da configuração espacial tanto no processo exibindo regularidade quanto no processo de agrupamentos.

3 MATERIAL E MÉTODOS

3.1 Simulação

Foram geradas 1000 realizações de configurações exibindo agrupamentos e regularidade. Cada realização apresentava 100 pontos distribuídos dentro de um plano quadrado com área unitária.

Para simular as realizações exibindo agrupamentos, foi utilizado o algoritmo de Neyman-Scott descrito no capítulo 2.4. Foram simuladas configurações com 2, 5, 10 e 20 agrupamentos e variabilidades radiais iguais a 0,05; 0,10; 0,15; 0,20; 0,25; 0,30; 0,40 e 0,50. Assim, foram geradas 32 configurações, representando todas as combinações de número de agrupamentos e variabilidades radiais.

Para simular as realizações com regularidade, foi utilizado o algoritmo SSI descrito no capítulo 2.5. Foram simuladas configurações com as seguintes distâncias mínimas entre os pontos: 0,010; 0,012; 0,014; 0,016; 0,018; 0,020; 0,030; 0,040; 0,050 e 0,060, perfazendo um total de 10 configurações com regularidade.

Em cada uma das realizações foram aplicados os testes descritos nos capítulos 2.8 e 2.9. Para obter as estatísticas de testes da função K de Ripley, foi utilizado o valor da distância $r_o = 0,25$ como o limite superior, tanto na integral da medida de Cramer-vonMises (equação 11), como na soma da medida de Kolmogorov (equação 12). O valor $r_o = 0,25$ é recomendado por Diggle (2002) para configurações pontuais geradas em áreas unitárias. Para obter o *valor-p* dos testes de hipótese, utilizou-se 100 simulações de Monte Carlo, atendendo o que é recomendado por Diggle (2002)

Para obter as estatísticas de testes da função H de Dirichlet, foi utilizado o valor da área $r_o = 0,1$ como o limite superior, tanto da medida de Cramer-

vonMises (equação 18), como na medida de Kolmogorov (equação 17). O valor $r_o = 0,1$ foi obtido empiricamente a partir de simulação.

O valor da estatística $\widehat{E}[\widehat{H}(r)]$ para os testes da função H de Dirichlet foi calculado a partir de 1000 simulações, com 100 pontos distribuídos em uma área unitária, sob a hipótese de completa aleatoriedade espacial.

Em todas as estatísticas de testes onde foi utilizada a distância de Cramer-vonMises, a integral foi calculada numericamente usando o método do trapézio. Para obter o poder de cada teste, foi registrado o número de vezes que o teste rejeitou a hipótese nula de completa aleatoriedade espacial tanto para as configurações com agrupamentos como para configurações com regularidade. O número de rejeições corretas é uma medida do poder do teste. Para rejeitar a hipótese nula foi utilizado um nível de significância $\alpha = 5\%$.

Todo o trabalho computacional foi realizado através do uso de funções desenvolvidas no *software* R (R DEVELOPMENT CORE TEAM, 2009). Estas funções estão disponíveis nos anexos (anexo 1 ao anexo 14).

3.2 Aplicação

Para exemplificar a aplicação dos testes, foram considerados os dados referentes à localização da amostra de vinte árvores de *Capaifera langsdorffi* Desf., popularmente conhecida por copaíba ou pau de óleo, localizadas no campus da Universidade Federal de Lavras (UFLA), Lavras, MG. Trata-se de um fragmento florestal que possui uma área de 5,83 ha, conhecido como “Matinha da UFLA”. A vegetação é classificada como Floresta Estacional Semidecidual Montana (floresta tropical subcadúcifolia), com dossel emergente (Fme), de acordo com o sistema de classificação da Fundação Instituto Brasileiro de Geografia e Estatística – IBGE (1993). Este fragmento está

localizado nas coordenadas 21°14'40''S e 44°57'50''W, a uma altitude de 925 m (LOUZADA, 2003).

A posição geográfica das vinte árvores foi obtida por meio de bússola e a distância entre as mesmas, por trena, a fim de obter as coordenadas cartesianas destes indivíduos. Este procedimento é conhecido como caminhamento. Os dados obtidos estão na Tabela 1.

Tabela 1 Coordenadas cartesianas (x, y) de cada árvore de copaíba no fragment florestal

<i>X</i>	<i>Y</i>
163,45	155,47
125,27	157,26
77,48	141,78
77,48	146,08
95,17	133,69
90,55	131,54
100,00	142,80
47,56	109,07
59,76	106,87
56,83	103,72
110,59	102,52
132,70	112,83
135,21	106,86
154,70	81,39
182,28	95,28
194,69	105,69
278,42	78,02
274,53	74,13
266,27	64,04
306,72	2,20

Para testar a hipótese nula de que as árvores de copaíba apresentam uma distribuição completamente aleatória dentro do fragmento florestal, foram aplicados os testes descritos nos capítulos 2.8 e 2.9. Para obter as estatísticas de testes da função K de Ripley, o valor da distância $r_o = 0,25 \times \text{área}$ foi utilizado como o limite superior tanto para a medida de Cramer-vonMises (equação 12), como para a medida de Kolmogorov (equação 11). Este valor de r_o é recomendado por Diggle (2002) para configurações pontuais geradas em áreas, aproximadamente, retangulares.

Para obter as estatísticas de testes da função H de Dirichlet, o valor da área $r_o = 0,1 \times \text{área}$ foi utilizado como o limite superior tanto para a medida de Cramer-vonMises (equação 18), como para a medida de Kolmogorov (equação 17). O valor de r_o foi obtido empiricamente a partir de simulação.

O valor $\hat{E}[\hat{H}(r)]$ utilizado nas estatísticas de testes da função H de Dirichlet foi calculado a partir de 1000 simulações, com vinte pontos em cada simulação, distribuídos em uma área de 5,83 ha, aproximadamente retangular, sob a hipótese de completa aleatoriedade espacial.

Em todas as estatísticas de testes em que a distância de Cramer-vonMises foi utilizada, a integral foi calculada numericamente, usando o método do trapézio. Também foram utilizadas 100 simulações de Monte-Carlo para a obtenção dos *valores-p*. A análise dos dados foi feita através da utilização de funções desenvolvidas no *software* R (R DEVELOPMENT CORE TEAM, 2009). Estas funções estão disponíveis nos anexos.

4 RESULTADOS E DISCUSSÃO

4.1 Simulação

Ao se estudar o padrão espacial é necessário distinguir entre o padrão espacial observado e o processo espacial que o gerou. Um mapa de pontos específico deve ser entendido como uma realização particular e única (o padrão espacial observado) de um processo espacial, resultante da interação de inúmeros processos de pontos que poderiam gerar infinitos padrões espaciais diferentes. Logicamente, todos os padrões gerados pelos mesmos processos deveriam apresentar características semelhantes. As funções F , G , K e J são formas usuais de descrever tais características conforme pode ser visto em Bailey e Gatrell (1995), Cressie (1991) e Diggle (2002).

Mais especificamente, existe o interesse explícito de verificar se a configuração espacial observada foi gerada por um processo estocástico espacial completamente aleatório. Assim, testar a hipótese nula de completa aleatoriedade espacial é o primeiro passo na análise de configurações pontuais. Para testar esta hipótese nula, foram desenvolvidos diversos testes e procedimentos gráficos baseados nas funções F , G , K e J . Estes testes estão descritos em detalhes em Diggle (2002).

Existem alguns trabalhos comparando o poder dos testes baseados nessas funções. Diggle (1979) e Thonnes e Van Lieshout (1999) mostram que uma típica hierarquia de poder (Po) para mapas com 100 pontos contra agregação é dada por $Po(F) > Po(K) > Po(G) > Po(J)$ e contra regularidade é dada por $Po(G) > Po(K) > Po(J) > Po(F)$. Scalon (2006) mostra que a hierarquia do poder para configurações com número reduzido de eventos (10 pontos) contra agregação é dada por $Po(K) > Po(G) > Po(F) > Po(J)$, enquanto contra regularidade é dado por $Po(G) > Po(J) > Po(K) > Po(F)$. Esses trabalhos

mostram duas coisas: (a) a necessidade de usar testes contra a hipótese de aleatoriedade espacial baseados em, pelo menos, duas funções. (b) no caso de usar uma única função, esta deve ser a função K , pois é competitiva nas mais diversas situações.

Esta dissertação apresenta uma metodologia composta de testes de hipóteses e um procedimento gráfico para avaliar a hipótese de completa aleatoriedade espacial, tendo como base as áreas dos polígonos de um mosaico de Dirichlet. Como a função K é competitiva para configurações de diferentes processos espaciais, ela foi utilizada como parâmetro de comparação em relação à metodologia baseada no mosaico de Dirichlet.

Na Tabela 2, encontram-se os resultados da análise do poder das funções K e H em configurações com a presença de agrupamentos. A primeira observação a ser feita na Tabela 2 é que o poder dos testes baseados nas duas funções é reduzido quando o número de agrupamentos e/ou a variabilidade radial dos agrupamentos aumenta. A distância de Komogorov parece produzir testes mais poderosos que a distância de Cramer-VonMises. Na presença de um pequeno número de agrupamentos com pequena variabilidade radial, ou seja, configurações com agrupamentos bem definidos, não existem diferenças entre os testes baseados na função K e D . Na presença de um grande número de agrupamentos com grande variabilidade radial, ou seja, configurações geradas por um processo de agrupamentos, porém próximos de um processo aleatório, os testes baseados na função K mostraram ser mais poderosos.

Tabela 2 Número de vezes que os testes baseados nas funções K e H identificaram corretamente a configuração com agrupamentos (poder) em 1000 realizações

Nº Clusters	σ	Função K		Função H	
		$L_{M(Kolmogov)}$	$L_{I(Cramer)}$	$H_{M(Kolmogov)}$	$H_{I(Cramer)}$
02	0.05	1000	1000	1000	1000
02	0.10	1000	1000	1000	1000
02	0.15	1000	1000	1000	1000
02	0.20	1000	1000	1000	1000
02	0.25	997	1000	870	860
02	0.30	947	770	690	670
02	0.40	784	543	560	540
02	0.50	773	412	438	410
05	0.05	1000	1000	1000	1000
05	0.10	1000	1000	1000	1000
05	0.15	997	1000	1000	990
05	0.20	951	940	830	790
05	0.25	876	870	690	620
05	0.30	709	770	380	330
05	0.40	627	554	296	250
05	0.50	523	469	197	185
10	0.05	1000	1000	1000	1000
10	0.10	1000	1000	1000	1000
10	0.15	970	980	900	860
10	0.20	910	870	650	620
10	0.25	760	720	440	350
10	0.30	631	510	330	280
10	0.40	452	414	260	210
10	0.50	326	290	170	123
20	0.05	900	1000	1000	1000
20	0.10	931	972	990	980
20	0.15	760	822	840	770
20	0.20	606	672	590	512
20	0.25	518	541	270	240
20	0.30	470	467	169	180
20	0.40	362	359	98	102
20	0.50	250	271	52	41

A título de ilustração, as Figuras 11-14 mostram gráficos comparativos dos testes, considerando configurações com dois, cinco, dez e vinte agrupamentos. As Figuras 11-14 mostram que o poder dos testes baseados nas duas funções é reduzido quando a variabilidade radial dos agrupamentos aumenta, ou seja, a configuração se aproxima da completa aleatoriedade espacial. Nas Figuras 11-14, observa-se também que os testes baseados na distância de Kolmogorov demonstra ser o mais poderoso. Finalmente, pode-se observar que, na presença de um grande número de agrupamentos, os testes baseados na função K tendem a ser mais poderosos do que os testes baseados na função H .

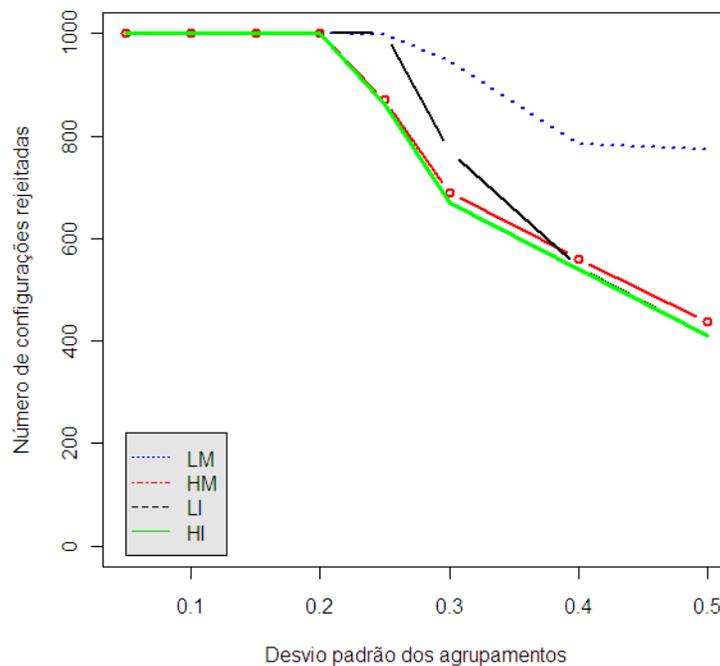


Figura 11 Poder dos testes em configurações com dois agrupamentos, em que LM é o teste baseado na função K , usando a distância de Kolmogorov; LI é o teste baseado na função K , usando a distância de Cramer-vonMises; HM é o teste baseado na função H , usando a distância de Kolmogorov; HI é o teste baseado na função H , usando a distância de Cramer-vonMises

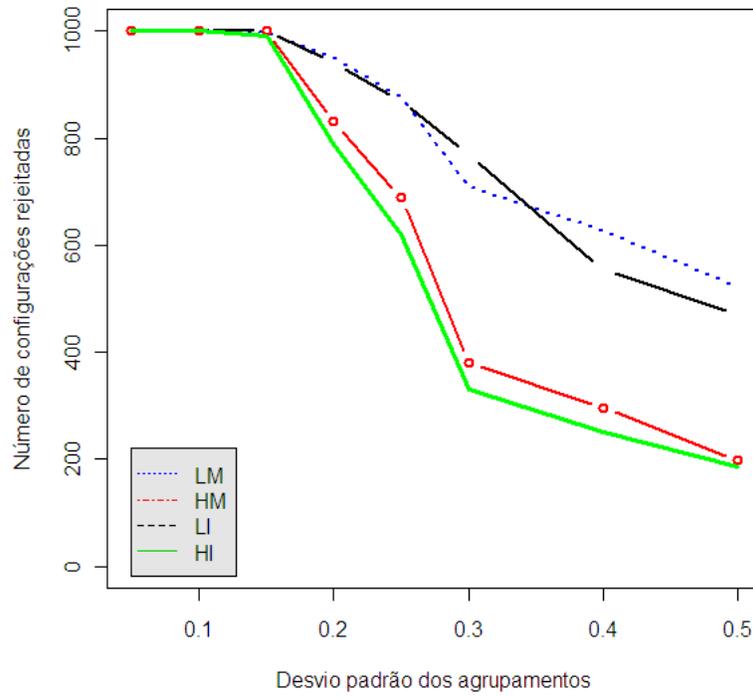


Figura 12 Poder dos testes em configurações com cinco agrupamentos, em que LM é o teste baseado na função K , usando a distância de Kolmogorov; LI é o teste baseado na função K , usando a distância de Cramer-vonMises; HM é o teste baseado na função H , usando a distância de Kolmogorov; HI é o teste baseado na função H , usando a distância de Cramer-vonMises

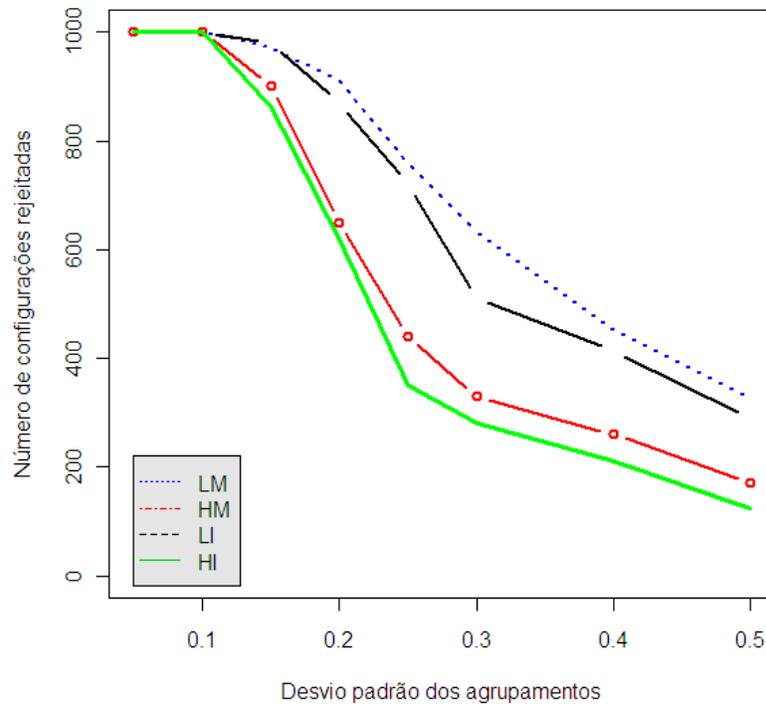


Figura 13 Poder dos testes em configurações com dez agrupamentos, em que LM é o teste baseado na função K , usando a distância de Kolmogorov; LI é o teste baseado na função K , usando a distância de Cramer-vonMises; HM é o teste baseado na função H , usando a distância de Kolmogorov; HI é o teste baseado na função H , usando a distância de Cramer-vonMises

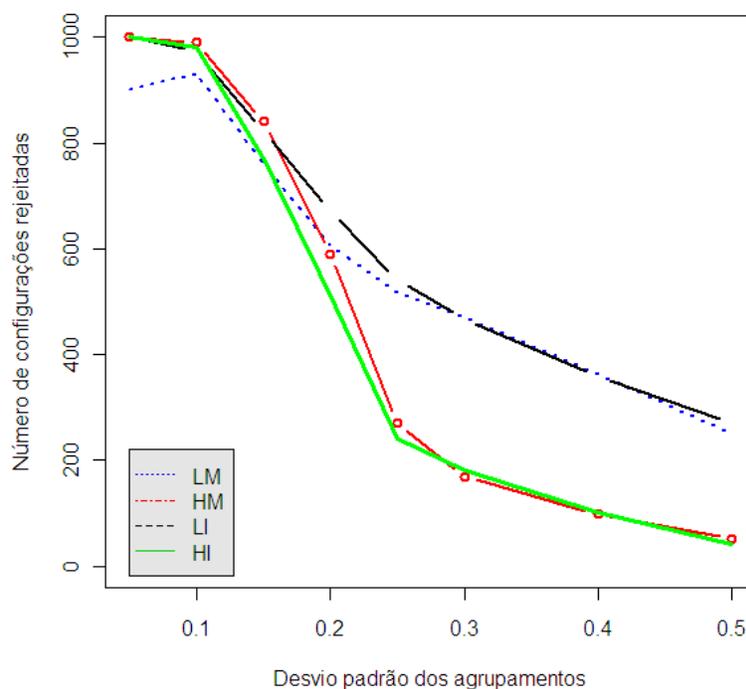


Figura 14 Poder dos testes em configurações com vinte agrupamentos, em que LM é o teste baseado na função K , usando a distância de Kolmogorov; LI é o teste baseado na função K , usando a distância de Cramer-vonMises; HM é o teste baseado na função H , usando a distância de Kolmogorov; HI é o teste baseado na função H , usando a distância de Cramer-vonMises

Os resultados obtidos nesta dissertação corroboram os resultados obtidos por Myles et al. (1995) quando demonstram que a função H é mais competitiva em relação à função K para detectar configurações com agrupamentos.

Na Tabela 3, são apresentados os resultados da análise do poder das funções K e H em configurações com a presença de regularidade.

Tabela 3 Número de vezes em que os testes baseados nas funções K e H identificaram corretamente a configuração com regularidade (poder) em 1000 realizações

Distância	H_M	H_I	L_M	L_I
0.010	30	46	130	49
0.012	39	48	256	41
0.014	44	29	513	53
0.016	43	41	896	79
0.018	54	43	996	83
0.020	38	34	1000	124
0.030	101	41	1000	882
0.040	313	115	1000	1000
0.050	867	410	1000	1000
0.060	1000	907	1000	1000

A Tabela 3 mostra que o poder dos testes baseados nas funções K e H aumentam quando a distância mínima entre os pontos também aumenta e que a distância de Komogorov leva a testes mais poderosos que a distância de Cramer-vonMises. A tabela 3 também deixa claro que existe uma superioridade no poder da função K sobre a função H para detectar configurações oriundas de um processo com regularidade.

A Figura 15 mostra gráficos comparativos do poder dos testes em configurações com regularidade.

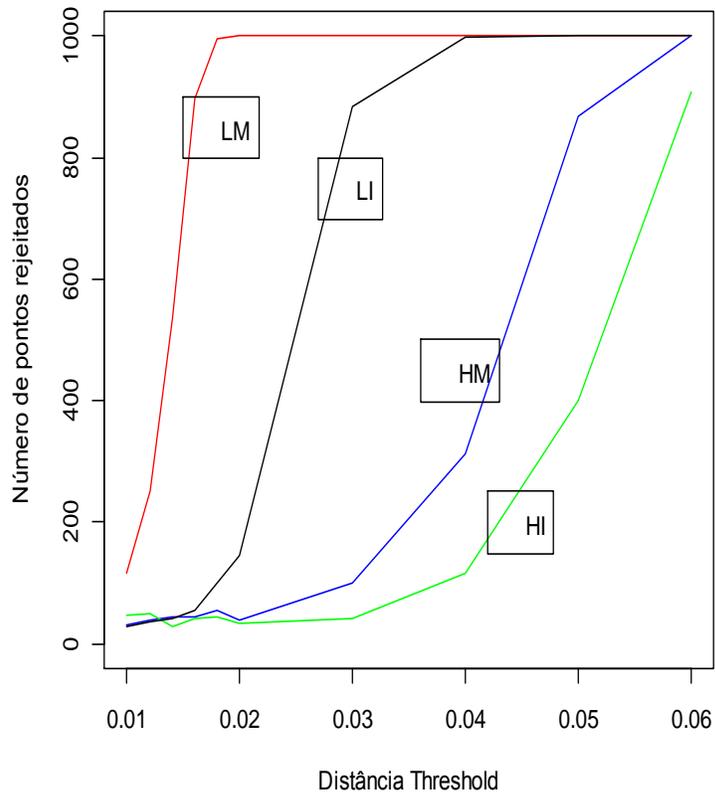


Figura 15 Poder dos testes em configurações com regularidade, em que LM é o teste baseado na função K , usando a distância de Kolmogorov; LI é o teste baseado na função K , usando a distância de Cramer-vonMises; HM é o teste baseado na função H , usando a distância de Kolmogorov; HI é o teste baseado na função H , usando a distância de Cramer-vonMises

A Figura 15 mostra que o poder dos testes baseados nas duas funções é reduzido com a diminuição da distância mínima entre os pontos, ou seja, quando a configuração se aproxima da completa aleatoriedade espacial. Observa-se também na Figura 15 que, dentro de cada função, os testes baseados

na distância de Kolmogorov se mostram mais poderosos que os testes baseados na distância de Cramer-von Mises.

Pode-se observar também que, a menos de configurações com grandes distâncias mínimas entre os pontos, onde os testes baseados nas funções H e K são equivalentes, os testes baseados na função K tendem a ser mais poderosos do que os testes baseados na função H .

Os resultados obtidos nesta dissertação corroboram os resultados obtidos por Myles et al. (1995) de que a função K é mais poderosa que a função H para detectar configurações exibindo regularidade.

Finalmente, deve-se ressaltar que, para uma análise completa da competitividade da função H perante a função K , o comportamento dos testes quanto ao erro do tipo I deveria ser investigado tendo por base essas funções. Poder-se-ia também comparar o poder e o erro do tipo I dos testes baseados na função H com o poder e o erro do tipo I dos testes baseados nas funções F , G e J .

4.2 Aplicação

O objetivo da análise foi testar a hipótese nula de que as árvores de copaíba apresentam uma distribuição completamente aleatória dentro do fragmento florestal. A Figura 16 mostra a posição das árvores de copaíba na região de estudo. Sendo que as coordenadas x e y são dadas em metros.

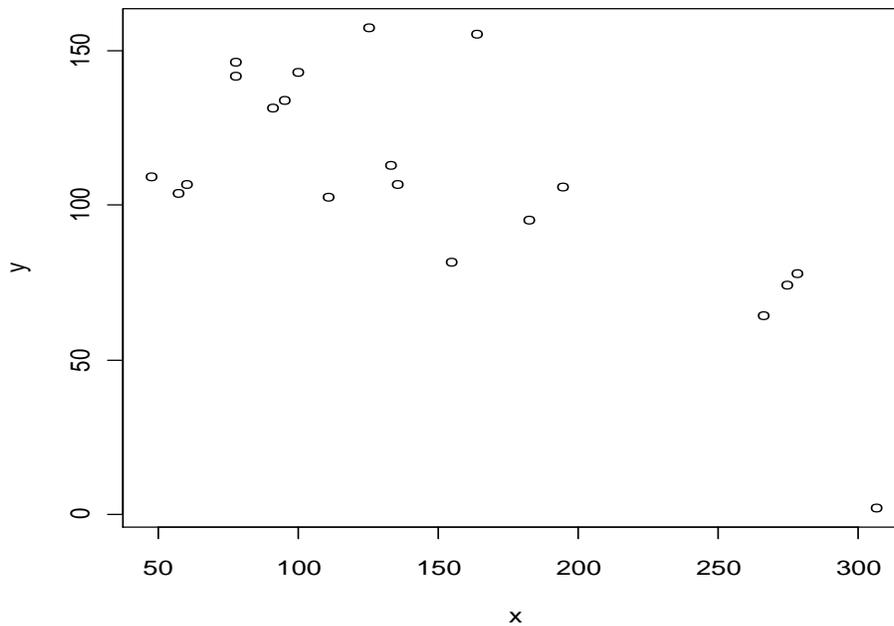


Figura 16 Posição das árvores de copaíba no fragmento florestal

A análise da Figura 16 não possibilita uma decisão quanto ao tipo de distribuição das árvores de copaíba. Assim, foram aplicados os testes de hipóteses descritos nos capítulos 2.8 e 2.9. Função H , utilizando a distância de Kolmogorov ($HM = 0.56$, $p = 0,1$); função H , utilizando a distância de Cramer-vonMises ($HI = 55$, $p = 0,1$).

Em função dos resultados dos testes de hipóteses (anexo 3), observa-se que todos os testes rejeitaram a hipótese nula de completa aleatoriedade espacial das árvores de copaíba no fragmento florestal. Entretanto, esses resultados não permitem avaliar a direção da rejeição da hipótese nula. Assim, conduziu-se a análise gráfica pelas funções L e H descritas, respectivamente, nos capítulos 2.8 e 2.9. Os resultados estão apresentados nas Figuras 17 e 18, onde é possível

observar que as duas funções levaram à rejeição da hipótese nula de completa aleatoriedade espacial em favor da existência de agrupamentos.

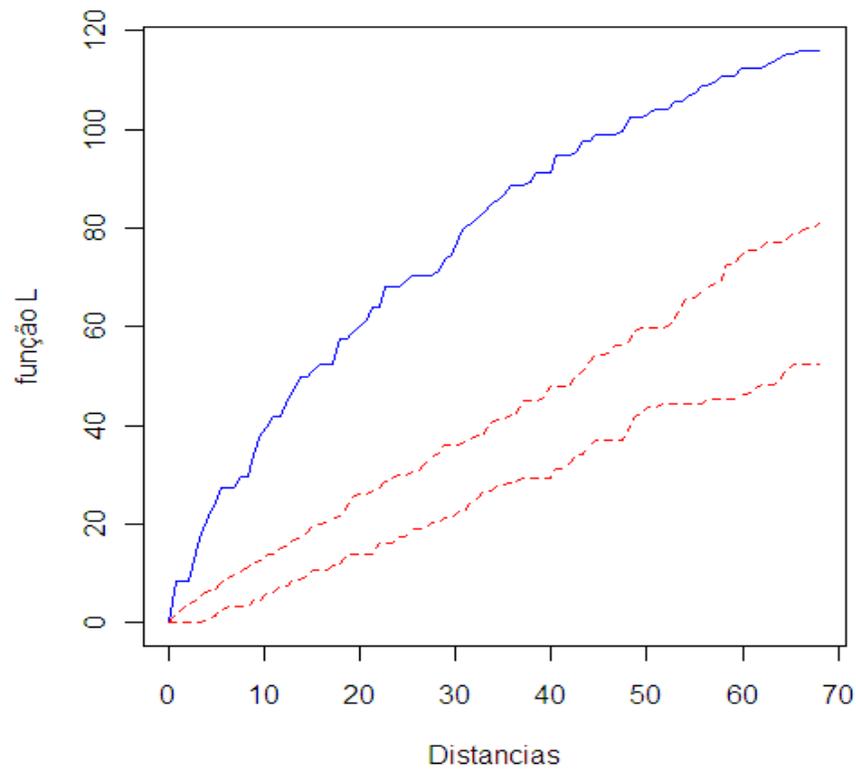


Figura 17 Função L observada (linha contínua em azul) e envelope de simulação com 99% de confiança (linhas tracejadas em vermelho), representado os limites da completa aleatoriedade espacial

As Figuras 17 e 18 também mostram que o gráfico da função L é mais explícito na detecção da não aleatoriedade. Entretanto, este fato pode ser devido à escala dos dois gráficos. Enquanto na função L, a escala é uma medida de distância, na função H , a escala é uma medida de área.

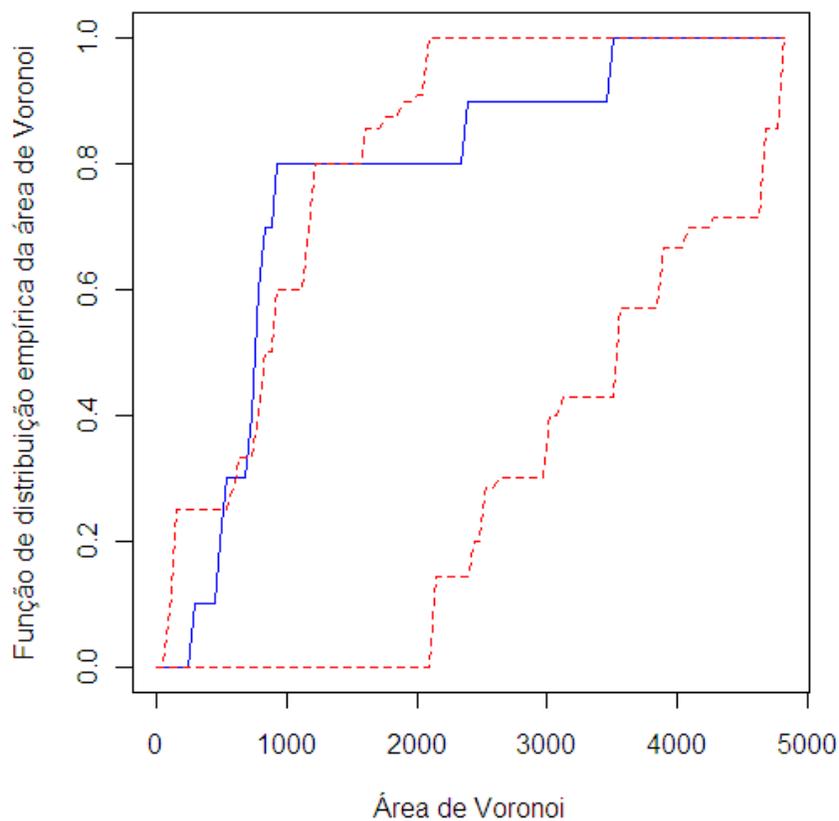


Figura 18 Função D observada (linha contínua em azul) e envelope de simulação com 99% de confiança (linhas tracejadas em vermelho) representado os limites da completa aleatoriedade espacial

O conjunto de dados analisado neste capítulo também foi investigado por Louzada (2003) e Olinda e Scalon (2010) com objetivos e metodologias diferentes das abordadas nesta dissertação.

Louzada (2003) usou o índice de Moran para verificar a existência de autocorrelação espacial dos marcadores genéticos das árvores. Os resultados obtidos mostraram a inexistência de autocorrelação espacial para a maioria dos marcadores. Nos casos em que ocorreu autocorrelação, esta foi positiva fraca.

Olinda e Scalon (2010) desenvolveram testes usando a função de correlação marcada para verificar a hipótese de independência entre marcas (marcadores genéticos) e pontos (localização das árvores). Esses autores mostraram que havia uma relação de dependência para alguns marcadores genéticos. Ambos os trabalhos verificaram que, para alguns marcadores genéticos, quanto mais próximas as árvores, mais próximos eram os marcadores. De maneira indireta, a presente dissertação corrobora os resultados obtidos por Louzada (2003) e Olinda e Scalon (2010). Apesar de as marcas do processo pontual não terem sido o principal foco de interesse desse trabalho, observou-se que a formação de agrupamentos das árvores de copáiba possivelmente se deve à fatores genéticos destas.

5 CONCLUSÃO

Os resultados obtidos nesta dissertação mostram que os testes baseados na função H de Dirichlet são competitivos com os testes baseados na função K de Ripley para detectar configurações geradas por processos pontuais com forte afastamento da hipótese nula de completa aleatoriedade espacial. Conforme os processos pontuais se aproximam da completa aleatoriedade espacial, verifica-se que os testes baseados na função K de Ripley são mais poderosos. Os testes baseados nas funções L e H levaram às mesmas conclusões quanto à presença de agrupamentos nas árvores de copaíbas no fragmento florestal da UFLA.

REFERÊNCIAS

BADDLEY, A. J.; TURNE, R. Spatstart an R Package for analysing spatial point patterns. **Journal of Statistical Software**, New York, v. 12, n. 6, June 2005.

BAILEY, T. C.; GATRELL, A. C. **Interactive spatial data analysis**. Essex: Longman Scientific, 1995

CAMERA, G. et al. **Introdução à ciências da geoinformação**. São Jose dos Campos: INPE, 2002.

CARMER, S. G; SWANSON, M. R. An evaluation of ten pairwise multiple comparison procedures by Monte Carlos methods. **Journal of the American Statistical Association**, Washington, v. 68, n. 341, p. 66-74, Mar. 1973.

CRESSIE, N. **Statistics for spatial data**. Chichester: J. Wiley, 1991.

DIGGLE, P. J. On parameters estimation and goodness-of-fit testing for spatial point patterns. **Biometrics**, Washington, v. 35, p. 87-101, 1979.

DIGGLE, P. J. **Statistical analysis of spatial point patterns**. 2nd ed. London: Arnold; New York: Oxford University, 2003

GATRELL, A. C. et al. **Spatial point pattern analysis application in geographical epidemiology**. Lonfon: Institute of the Institute of British Geographers, 1995.

GREEN, P. J.; SIBSON, R. Computing dirichlet tessellations in the plane. **The Computer Journal**, Oxford, v. 21, p. 168-173, 1978.

HOCHBERG, Y.; TAMHANE, A. C. **Multiple comparison procedures**. New York: J. Wiley, 1987

LIMA, R. R. **Modelagem espaço-temporal para dados de incidência para doenças de plantas**. 2005. 149 f. Tese (Doutorado em Agronomia/Estatística e Experimentação Agronômica) – Universidade de São Paulo, Escola Superior de Agricultura “Luiz de Queiroz”, Piracicaba, 2005.

LOUZADA, J. M. **Avaliação e aplicação de teste para a detecção de autocorrelação espacial utilizando dados de marcadores genéticos**. 2003. --- f. Dissertação (Mestrado em Estatística e Experimentação Agropecuária) – Universidade Federal de Lavras, Lavras, 2003.

KENKEL, N. C.; HOSKINS, J. A.; HOSKINS, W. D. Edge effects in the use of area polygons to study competition. **Ecology**, Columbus, v. 70, p. 272-274, 1989.

MEYER, P. L. **Probabilidade**: aplicações à estatística. Tradução do Prof. Ruy de C.B. Lourenço Filho. 2. ed. Rio de Janeiro: Livros Técnicos e Científicos, 1983.

MOOD, A. M.; GRAYBILL, F. A.; BOES, D. C. **Introduction to the theory of statistics**. 3. ed. New York: J. Wiley, 1974.564 p.

MYLES, J. P. et al. Statistical tests for clustering of second phases in composite materials. **Philosophical Magazine A**, London, v. 72, n. 2, p. 515-528, 1995.

OLINDA, R. A.; SCALON, J. D. Métodos de Monte Carlos para análise de processos pontuais marcados. **Revista Brasileira de Biometria**, São Paulo, v. 28, n. 1, p. 39-56, 2010.

R DEVELOPMENT CORE TEAM. R: a language and environment for statistical computing. Viena: R Foundation for Statistical Computing, 2009. Disponível em: <<http://www.R-project.org>>. Acesso em: 15 jul. 2009.

RIPLEY, B. D. Modelling spatial patterns (with discussion). **Journal of the Royal Statistical Society**, Oxford, v. 39, n. 2, p. 172-212, June 1977.

RIPLEY, B. D. **Spatial statistics**. London: J. Wiley, 1981.

RIPLEY, B. D. The second-order analysis of stationary point patterns. **Journal of Applied Probability**, Sheffield, v. 13, p. 255-266, 1976.

SCALON, J. D. Identificação de padrões espaciais em distribuições de casos de doenças. **Vertentes**, São João Del Rei, v. 14, p.7-14, 1999.

SCALON, J. D. et al. Spatial pattern analysis of second-phase particles in composite materials. **Material Science and Engineering A – Structural Materials Properties Microstructure and Processing**, Lausanne, v. 356, p. 245-257, 2003.

STOYAN, D.; STOYAN, H. **Fractals, random shapes and point fields**. New York: J. Willey, 1994.

UPTON, G.; FINGLETON, B. **Spatial data analysis by example**. New York: J. Willey, 1985. v. 1.

WALTER, L. A.; GOTAWAY, C. A. **Applied spatial statistics for public health data**. New Jersey: J. Wiley, 2004. 641 p. (Probability and statistics).

WILKINSON, I. **SYSGRAPH**: the system for graphics. Evanston: SYSTAT, 1990.

VAN LIESHOUT, M. N. M.; BADDELEY, A. J. A nonparametric measure of spatial interaction in point patterns. **Statistica Neerlandica**, Cambridge, v. 50, p. 344-361, 1996.

ANEXOS

ANEXO A	Programa CLUSGEN Esta função gera configurações com agrupamentos	68
ANEXO B	Programa Denvelopes Esta função implementa simulações de Monte-Carlo e cria envelopes de simulação para testar a hipótese de completa aleatoriedade espacial (CSR) usando a função de Dirichlet.....	69
ANEXO C	Programa Dtest Esta função implementa os testes Monte-Carlo, baseados na função de Dirichlet para testar a hipótese de completa aleatoriedade espacial (CSR).....	71
ANEXO D	Programa ehrD Esta função implementa simulações de Monte-Carlo para obter $E[H(r)]$	74
ANEXO E	Programa powerDLUSTER Esta função tem a finalidade de verificar o poder do teste de Dirichlet para configurações com agrupamentos.....	75
ANEXO F	Programa powerDSSI Esta função tem a finalidade de verificar o poder do teste de Dirichlet para configurações com regularidade.....	77
ANEXO G	Programa powerKCLUSTER Esta função tem a finalidade de verificar o poder do teste da função K de Ripley para agrupamentos.....	79
ANEXO H	Programa powerKSSI Esta função tem a finalidade de verificar o poder do teste da função K de Ripley para regularidade.....	81
ANEXO I	Programa simulaDCLUSTER Esta função é utilizada	

	dentro da função powerDCLUSTER.....	83
ANEXO J	Programa simulaDSSI Esta função é utilizada dentro da função powerDSSI.....	86
ANEXO L	Programa simulaKCLUSTER Esta função é utilizada dentro da função powerKCLUSTER.....	89
ANEXO M	Programa simulaKSSI Esta função é utilizada dentro da função powerKSSI.....	91
ANEXO N	Programa Kenvelopes Esta função implementa simulações de Monte-Carlo para obter envelopes de simulação para testar a hipótese de completa aleatoriedade espacial(CSR) usando a função K de Ripley.....	93

ANEXO A

PROGRAMA CLUSGEN

```
function(ncen,npoi,sd)
{
centers<-csr(poly=matrix(c(0,0,1,1,0,0,1,1,0,0),5,2),ncen)
np<-floor(1+ncen*runif(npoi))
xydis<-spoints(rnorm(npoi*2,0,sd))
points<-centers[np,]+xydis
list(Centers=centers,points=points)
}
```

,

ANEXO B

Programa Denvelopes

```

function(event, nsim = 10, ma=0.1, lengt=100){
#
# This function implements monte-carlo simulations
# to create and plot simulation envelopes to test
# against the hypothesis of complete spatatial randomness (CSR)
# of an observed spatial point pattern.
# It uses the area of each Voronoi polygon.
#
# OBS: You need Spatstat and tripack loaded!
#
# Where:
# event: ppp class object that contains the coordinates of events and
marks
# nsim: number of monte-carlo simulations
# lengt: number of classes of areas
# ma: Percentual value of the maximum size of area used in calculation
(0<ma<1)
#
# This is crude code by J.D.Scalon. No warranty! (02/05/10)
#
# starting values
#
  np<- length(event$x)
  ma<-ma*(max(event$x)*max(event$y))
#
# Get the empirical distribution function (EDF) of the observed areas of
the Voronoi polygons
#
  data.vm<-voronoi.mosaic(event$x,event$y)
  data.area<-voronoi.area(data.vm)
  r<-seq(0,ma,length=lengt+1)
  rr<-seq(0,ma,length=lengt)
  data.h<-hist(data.area[data.area<ma],breaks=r,plot=FALSE)
  data.counts<-cumsum(data.h$count)
  ko<- data.counts/sum(data.h$count)
#
#
# Doing Monte-Carlo simulation under the CSR

```

```

#
hold<-matrix(0,nsim,length)
for(i in (1:nsim)) {
  x<-runif(np,0,max(event$x))
  y<-runif(np,0,max(event$y))
  xy<-cbind(x,y)
  spts<- as.ppp(xy, c(0,max(event$x),0,max(event$y)))
data.vm1<-voronoi.mosaic(spts)
data.area1<-voronoi.area(data.vm1)
r1<-seq(0,ma,length=length+1)
data.h1<-hist(data.area1[data.area1<ma],breaks=r1,plot=FALSE)
data.counts1<-cumsum(data.h1$counts)
k<- matrix(data.counts1/sum(data.h1$counts),1,length)
  hold[i,]<-k
}
#
# Observe that each row of hold contains the EDF for a single
# CSR simulation on the area
#
mn<- apply(hold,2,mean)
up<- apply(hold,2,max)
dow<- apply(hold,2,min)
#
#
# Creating plot
#
#
y<-seq (0,1,length=length(r))
par(pty = "s")
plot(r, y, type = "n", xlab = "Voronoi Areas", ylab = "Empirical
Distribution Function of the Voronoi area")
lines (rr, ko, col="blue")
lines (rr, up, lty=2, col="red")
lines (rr, dow,lty=2,col="red")
}

```

ANEXO C

Programa Dteste

```

function(event, nsim = 10, ma = 0.2, nsimEH=100, lengt=100){
#
# This function implements monte-carlo simulations
# for testing against the hypothesis of complete spatial
# randomness (CSR)of an observed spatial point pattern.
#
# It uses the area of each Voronoi polygon.
#
# Cramer-vonMises and Kolmogorov measure of distances are used for
# getting statistics of tests.
#
# OBS: You need Spatstat and tripack loaded!
#
# Where:
# event: ppp class object that contains the coordinates of events
# nsim: number of monte-carlo simulations
# nsimEH: number of monte-carlo simulations for getting E[H(r)]
# lengt: number of classes of areas
# ma: Percentual value of the maximum size of area used in calculation
(0<ma<1)
#
# This is crude code by J.D.Scalon. No warranty! (02/05/10)
#
# starting values
#
  np<- length(event$x)
  ma<-ma*(max(event$x)*max(event$y))
#
# Get the distribution of the observed areas of the Voronoi polygons
#
  data.vm<-voronoi.mosaic(event$x,event$y)
  data.area<-voronoi.area(data.vm)
  r<-seq(0,ma,length=lengt+1)
  data.h<-hist(data.area[data.area<ma],breaks=r,plot=FALSE)
  data.counts<-cumsum(data.h$count)
  ko<- data.counts/sum(data.h$count)
#
# Doing Monte-Carlo simulation under CSR for getting E[H(r)]

```

```

#
hold<-matrix(0,nsimEH,lengt)
for(i in (1:nsimEH)) {
  x<-runif(np,0,max(event$x))
  y<-runif(np,0,max(event$y))
  xy<-cbind(x,y)
  spts<- as.ppp(xy, c(0,max(event$x),0,max(event$y)))
  data.vm1<-voronoi.mosaic(spts)
  data.area1<-voronoi.area(data.vm1)
  r1<-seq(0,ma,length=lengt+1)
  data.h1<-hist(data.area1[data.area1<ma],breaks=r1,plot=FALSE)
  data.counts1<-cumsum(data.h1$counts)
  k<- matrix(data.counts1/sum(data.h1$counts),1,lengt)
  hold[i,]<-k
}
#
# Observe that each row of hold contains the EDF for a single
# CSR simulation on the area.
#
# kotheo is E[H(r)]
#
  kotheo<- apply(hold,2,mean)
#
# Get the integral for the observed spatial point pattern
#
  lent <- length(r)
  D<-(ko-kotheo)^2
  step<-max(r)/lent
  integ<-0
  steplength <- step * 2
  for(count in (2:(lent - 2))) {
    integ <- integ + (D[count - 1] + 4 * D[count] + D[count + 1]) *
      steplength
  }
  integ <- integ/6
#
# Get the kolmogorov distance for the observed spatial point pattern
#
  DK<-max (abs (ko-kotheo))
#
# Doing Monte-Carlo simulation under the CSR
#

```

```

for (i in (2:nsim)) {
  x<-runif (np,0, max(event$x))
  y<-runif (np,0, max(event$y))
  xy<-cbind(x,y)
  spts<- as.ppp(xy, c(0,max(event$x),0,max(event$y)))
  data.vm2<-voronoi.mosaic(spts)
  data.area2<-voronoi.area(data.vm2)
  r2<-seq(0,ma,length=length+1)
  data.h2<-hist(data.area2[data.area2<ma],breaks=r2,plot=FALSE)
  data.counts2<-cumsum (data.h2$counts)
  k<- data.counts2/sum (data.h2$counts)
  #
  # Get the integral under CSR
  #
  lent <- length(r1)
  t <- (k - kotheo)^2
  teg <- 0
  for(count in (2:(lent - 2))) {
    teg <- teg + (t[count - 1] + 4 * t[count] + t[count + 1]) *
      steplength
  }
  teg <- teg/6
  integ <- c(integ, teg)
  #
  # Get kolmogorov distance
  #
  test<-max(abs(k-kotheo))
  DK<-c (DK, test)
  }
  pi <- rank(integ)[1]
  pk <- rank(DK)[1]
  pvaluei <- (nsim + 1 - pi)/nsim
  pvaluek <- (nsim + 1 - pk)/nsim
  #
  list (Est_Cramer = integ[1], p_value_Cramer = pvaluei, Est_Kolm =
DK[1], p_value_Kolm = pvaluek)
}

```

ANEXO D

Programa ehrD

```

function (np=100, ma = 0.2, nsimEH=1000, lengt=100){
#
# This function implements monte-carlo simulations under CSR for
getting E[H(r)]
#
# Where:
# nsimEH: number of monte-carlo simulations for getting E[H(r)]
# lengt: number of classes of areas
# np: number of events
# ma: Percentual value of the maximum size of area used in calculation
(0<ma<1)
#
# OBS: You need Spatstat and tripack loaded!
#
# This is crude code by J.D.Scalon. No warranty! (09/04/10)
#
#
  hold<-matrix(0,nsimEH,lengt)
  for (i in (1:nsimEH)) {
    x<-runif (np,0,1)
    y<-runif (np,0,1)
    xy<-cbind(x,y)
    spts<- as.ppp(xy,c(0,1,0,1))
    data.vm1<-voronoi.mosaic(spts)
    data.area1<-voronoi.area(data.vm1)
    r1<-seq(0,ma,length=lengt+1)
    data.h1<-hist (data.area1[data.area1<ma],breaks=r1,plot=FALSE)
    data.counts1<-cumsum (data.h1$counts)
    k<- matrix (data.counts1/sum(data.h1$counts),1,lengt)
    hold [i,]<-k
  }
#
  kotheo<- apply(hold,2,mean) # kotheo is E[H(r)]
#
return(kotheo=kotheo)

```

ANEXO E

Programa powerDCLUSTER

```

function (p=0.05, pv=10){
#
# This function implements monte-carlo simulations
# for getting the power of a test against the hypothesis of complete
spatial
# randomness (CSR)of an observed spatial point pattern.
# Point patterns have area equal to one with 100 events (points)
#
# The test uses the area of each Voronoi polygon.
# Cramer-vonMises and Kolmogorov measure of distances are used for
# getting statistics of tests and calls the function simula.
#
# OBS: You need Spatstat and tripack loaded!
#
# Where:
# pv: number of simulations for getting power
# p: cutoff point
#
# This is crude code by J.D.Scalon. No warranty! (09/04/10)
#
# setting parameters
#
  clusters<-rep(c(2,5,10,20), each=6)# number of clusters
  sigma<-rep(c(0.05,0.10,0.15,0.20,0.25,0.30),    times=4)#    radial
variability
  pontos<-rep(c(140),each=24)
#
#
  eventos<-cbind(clusters, sigma, pontos)
#
# simulating clusters
#
  contK<-rep(0,nrow(eventos))
  contC<-rep(0,nrow(eventos))
#
# getting power
#
  for(i in 1:nrow(eventos)){

```

```
      for(j in 1:pv){
ze<-clusgen(eventos[i,1],eventos[i,3],eventos[i,2])
x<-ze$points[,1]
  y<-ze$points[,2]
X<-ppp(x,y,c(0,1),c(0,1))
a<-simulaDCLUSTER(event=X)
ptesK<-a$p_valueKolm
ptesC<-a$p_valueCramer
if(ptesK<p) contK[i]<-contK[i] +1
if(ptesC<p) contC[i]<-contC[i] +1
}
}
result<-cbind(eventos, contK, contC)
list(resultados = result)
}
```

ANEXO F

Programa powerDSSI

```

function (p=0.05, pv=100){
#
# This function implements monte-carlo simulations
# for getting the power of a test against the hypothesis of complete
spatial
# randomness (CSR)of an observed spatial point pattern.
# Point patterns have area equal to one with 100 events (points)
#
# The test uses the area of each Voronoi polygon.
# Cramer-vonMises and Kolmogorov measure of distances are used for
# getting statistics of tests and calls the function simula.
#
# OBS: You need spatstat and tripack loaded!
#
# Where:
# pv: number of simulations for getting power
# p: cutoff point
#
# This is crude code by J.D.Scalon. No warranty! (09/04/10)
#
# setting parameters
#
  dist<-c(0.01, 0.012, 0.014, 0.016, 0.018, 0.020, 0.030, 0.040, 0.050,
0.060)# inhibition distance
  pontos<-rep(c(100),each=10)
#
  eventos<-cbind(dist, pontos)
#
# simulating clusters
#
  contK<-rep(0,nrow(eventos))
  contC<-rep(0,nrow(eventos))
#
# getting power
#
  for(i in 1:nrow(eventos)){
    for(j in 1:pv){
      X<-rSSI(r=eventos[i,1],n=eventos[i,2])

```

```
a<-simulaDSSI(event=X)
ptesK<-a$p_valueKolm
ptesC<-a$p_valueCramer
if(ptesK<p) contK[i]<-contK[i] +1
if(ptesC<p) contC[i]<-contC[i] +1
}
}
result<-cbind(eventos, contK, contC)
list(resultados = result)
}
```

ANEXO G

Programa powerKCLUSTER

```

function (p=0.05, pv=2){
#
# This function implements monte-carlo simulations
# for getting the power of a test against the hypothesis of complete
spatial
# randomness (CSR)of an observed spatial point pattern.
# Point patterns have area equal to one with 100 events (points)
#
# The test uses the K-function by Ripley.
# Cramer-vonMises and Kolmogorov measure of distances are used for
# getting statistics of tests and calls the function simula.
#
# OBS: You need Spatstat and splancs loaded!
#
# Where:
# pv: number of simulations for getting power
# p: cutoff point
#
# This is crude code by J.D.Scalon. No warranty! (09/04/10)
#
# setting parameters
#
#
clusters<-rep(c(2,5,10,20), each=6)# number of clusters
sigma<-rep(c(0.05,0.10,0.15,0.20,0.25,0.30), times=4)# radial
variability
pontos<-rep(c(140),each=24)
#
eventos<-cbind(clusters, sigma, pontos)
#
# simulating clusters
#
contK<-rep(0,nrow(eventos))
contC<-rep(0,nrow(eventos))
#
# getting power
#
for (i in 1:nrow(eventos)){

```

```
      for (j in 1:pv){
ze<-clusgen(eventos[i,1],eventos[i,3],eventos[i,2])
x<-ze$points[,1]
  y<-ze$points[,2]
X<-ppp(x,y,c(0,1),c(0,1))
a<-simulaKCLUSTER(event=X)
ptesK<-a$p_valueKolm
ptesC<-a$p_valueCramer
if(ptesK<p) contK[i]<-contK[i] +1
if(ptesC<p) contC[i]<-contC[i] +1
}
}
result<-cbind(eventos, contK, contC)
list(resultados = result)
}
```

ANEXO H

Programa powerKSSI

```

function(p=0.05, pv=100){
#
# This function implements monte-carlo simulations
# for getting the power of a test against the hypothesis of complete
spatial
# randomness (CSR)of an observed spatial point pattern.
# Point patterns have area equal to one with 100 events (points)
#
# Inhibition is the alternative hypothesis
#
# The test uses the K-function (actually L-function) by Ripley.
# Cramer-vonMises and Kolmogorov measure of distances are used for
# getting statistics of tests and calls the function simula.
#
# OBS: You need Spatstat loaded!
#
# Where:
# pv: number of simulations for getting power
# p: cutoff point
#
# This is crude code by J.D.Scalon. No warranty! (08/04/10)
#
# setting parameters
#
#
  dist<-c(0.01, 0.012, 0.014, 0.016, 0.018, 0.020, 0.030, 0.040, 0.050,
0.060)# inhibition distance
  pontos<-rep(c(100),each=10)
#
  eventos<-cbind(dist, pontos)
#
  contK<-rep(0,nrow(eventos))
  contC<-rep(0,nrow(eventos))
#
# getting power
#
  for(i in 1:nrow(eventos)){
    for(j in 1:pv){

```

```
X<-rSSI(r=eventos[i,1],n=eventos[i,2])
a<-simulaKSSI(event=X)
ptesK<-a$p_valueKolm
ptesC<-a$p_valueCramer
if(ptesK<p) contK[i]<-contK[i] +1
if(ptesC<p) contC[i]<-contC[i] +1
}
}
result<-cbind(eventos, contK, contC)
list(resultados = result)
}
```

ANEXO I

Programa simulaDCLUSTER

```

function(event, kotheo=dataD, nsim = 100, np=100, ma = 0.2,
lengt=100){
  #
  # This function implements monte-carlo simulations
  # for getting the power of a test against the hypothesis of complete
spatial
  # randomness (CSR) of an observed spatial point pattern.
  # Point patterns have area equal to one with 100 events (points)
  #
  # The test uses the area of each Voronoi polygon.
  # Cramer-von Mises and Kolmogorov measure of distances are used for
  # getting statistics of tests.
  #
  # OBS: You need Spatstat and tripack loaded!
  #
  # Where:
  # kotheo: E[H(r)] obtained by using the function ehr
  # nsim: number of monte-carlo simulations
  # lengt: number of classes of areas
  # np: number of events
  # ma: Percentual value of the maximum size of area used in calculation
(0<ma<1)
  #
  # This is crude code by J.D.Scalon. No warranty! (09/04/10)
  #
  # Get the distribution of the observed areas of the Voronoi polygons
  #
  data.vm<-voronoi.mosaic(event$x,event$y)
  data.area<-voronoi.area(data.vm)
  r<-seq(0,ma,length=lengt+1)
  data.h<-hist(data.area[data.area<ma],breaks=r,plot=FALSE)
  data.counts<-cumsum(data.h$count)
  ko<- data.counts/sum(data.h$count)
  #
  #
  # Get the integral for the observed spatial point pattern
  #
  lent <- length(r)

```

```

D<-(ko-kotheo)^2
step<-max(r)/lent
integ<-0
steplength <- step * 2
for(count in (2:(lent - 2))) {
  integ <- integ + (D[count - 1] + 4 * D[count] + D[count + 1]) *
  steplength
}
integ <- integ/6
#
# Get the kolmogorov distance for the observed spatial point pattern
#
DK<-max(abs(ko-kotheo))
#
# Doing Monte-Carlo simulation under the CSR
#
for(i in (2:nsim)) {
  x<-runif(np,0,1)
  y<-runif(np,0,1)
  xy<-cbind(x,y)
  spts<- as.ppp(xy, c(0,1,0,1))
  data.vm2<-voronoi.mosaic(spts)
  data.area2<-voronoi.area(data.vm2)
  r2<-seq(0,ma,length=lengt+1)
  data.h2<-hist(data.area2[data.area2<ma],breaks=r2,plot=FALSE)
  data.counts2<-cumsum(data.h2$counts)
k<- data.counts2/sum(data.h2$counts)
#
# Get the integral under CSR
#
lent <- length(r)
t <- (k - kotheo)^2
teg <- 0
for(count in (2:(lent - 2))) {
  teg <- teg + (t[count - 1] + 4 * t[count] + t[count + 1]) *
  steplength
}
teg <- teg/6
integ <- c(integ, teg)
#
# Get kolmogorov distance under CSR
#

```

```
test<-max(abs(k-kotheo))
DK<-c(DK,test)
}
pi <- rank(integ)[1]
pk <- rank(DK)[1]
pvaluei <- (nsim + 1 - pi)/nsim
pvaluek <- (nsim + 1 - pk)/nsim
list(p_valueKolm=pvaluek, p_valueCramer=pvaluei)
}
```

ANEXO J

Programa simulaDSSI

```

function(event, kotheo=dataD, nsim = 100, np=100, ma = 0.2,
lengt=100){
  #
  # This function implements monte-carlo simulations
  # for getting the power of a test against the hypothesis of complete
spatial
  # randomness (CSR) of an observed spatial point pattern.
  # Point patterns have area equal to one with 100 events (points)
  #
  # The test uses the area of each Voronoi polygon.
  # Cramer-vonMises and Kolmogorov measure of distances are used for
  # getting statistics of tests.
  #
  # OBS: You need Spatstat and tripack loaded!
  #
  # Where:
  # kotheo: E[H(r)] obtained by using the function ehr
  # nsim: number of monte-carlo simulations
  # lengt: number of classes of areas
  # np: number of events
  # ma: Percentual value of the maximum size of area used in calculation
(0<ma<1)
  #
  # This is crude code by J.D.Scalon. No warranty! (09/04/10)
  #
  # Get the distribution of the observed areas of the Voronoi polygons
  #
  data.vm<-voronoi.mosaic(event$x,event$y)
  data.area<-voronoi.area(data.vm)
  r<-seq(0,ma,length=lengt+1)
  data.h<-hist(data.area[data.area<ma],breaks=r,plot=FALSE)
  data.counts<-cumsum(data.h$count)
  ko<- data.counts/sum(data.h$count)
  #
  #
  # Get the integral for the observed spatial point pattern
  #
  lent <- length(r)

```

```

D<-(ko-kotheo)^2
step<-max(r)/lent
integ<-0
steplength <- step * 2
for(count in (2:(lent - 2))) {
  integ <- integ + (D[count - 1] + 4 * D[count] + D[count + 1]) *
  steplength
}
integ <- integ/6
#
# Get the kolmogorov distance for the observed spatial point pattern
#
DK<-max(abs(ko-kotheo))
#
# Doing Monte-Carlo simulation under the CSR
#
for(i in (2:nsim)) {
  x<-runif(np,0,1)
  y<-runif(np,0,1)
  xy<-cbind(x,y)
  spts<- as.ppp(xy, c(0,1,0,1))
  data.vm2<-voronoi.mosaic(spts)
  data.area2<-voronoi.area(data.vm2)
  r2<-seq(0,ma,length=length+1)
  data.h2<-hist(data.area2[data.area2<ma],breaks=r2,plot=FALSE)
  data.counts2<-cumsum(data.h2$counts)
k<- data.counts2/sum(data.h2$counts)
#
# Get the integral under CSR
#
lent <- length(r)
t <- (k - kotheo)^2
teg <- 0
for(count in (2:(lent - 2))) {
  teg <- teg + (t[count - 1] + 4 * t[count] + t[count + 1]) *
  steplength
}
teg <- teg/6
integ <- c(integ, teg)
#
# Get kolmogorov distance under CSR
#

```

```
test<-max(abs(k-kotheo))
DK<-c(DK,test)
}
pi <- rank(integ)[1]
pk <- rank(DK)[1]
pvaluei <- (nsim + 1 - pi)/nsim
pvaluek <- (nsim + 1 - pk)/nsim
list(p_valueKolm=pvaluek, p_valueCramer=pvaluei)
}
```

ANEXO L

Programa simulaKCLUSTER

```

function(event, nsim = 100, np=100){
#
# This function implements monte-carlo simulations
# for getting the power of a test against the hypothesis of complete
spatial
# randomness (CSR)of an observed spatial point pattern.
# Point patterns have area equal to one with 100 events (points)
#
# The test uses K-function by Ripley.
# Cramer-vonMises and Kolmogorov measure of distances are used for
# getting statistics of tests.
#
# OBS: You need Spatstat and splancs!
#
# Where:
# nsim: number of monte-carlo simulations
# np: number of events
#
# This is crude code by J.D.Scalon. No warranty! (09/04/10)
#
# Get the distribution of the observed K-function
#
  r<-seq(0,0.25,length=200)
k<- Kest(event, r=r, correction="Ripley")
  ko<-sqrt(k$iso/pi)
#
# Get the integral for the observed spatial point pattern
#
  lent<- length(r)
  D<-(ko-r)^2
  step<-max(r)/lent
  integ<-0
  steplength <- step * 2
  for(count in (2:(lent - 2))) {
    integ <- integ + (D[count - 1] + 4 * D[count] + D[count + 1]) *
      steplength
  }
  integ <- integ/6

```

```

#
# Get the kolmogorov distance for the observed spatial point pattern
#
  DK<-max(abs(ko-r))
#
# Doing Monte-Carlo simulation under the CSR
#
  for(i in (2:nsim)) {
    x<-runif(np,0,1)
    y<-runif(np,0,1)
    xy<-cbind(x,y)
    spts<- as.ppp(xy, c(0,1,0,1))
    kk<- Kest(spts, r=r, correction="Ripley")
    kl<-sqrt(kk$iso/pi)
#
# Get the integral under CSR
#
    lent <- length(r)
    t <- (kl - r)^2
    step<-max(r)/lent
    teg <- 0
    steplength <- step * 2
    for(count in (2:(lent - 2))) {
      teg <- teg + (t[count - 1] + 4 * t[count] + t[count + 1]) *
        steplength
    }
    teg <- teg/6
    integ <- c(integ, teg)
#
# Get kolmogorov distance under CSR
#
    test<-max(abs(kl-r))
    DK<-c(DK,test)
  }
pi <- rank(integ)[1]
pk <- rank(DK)[1]
pvaluei <- (nsim + 1 - pi)/nsim
pvaluek <- (nsim + 1 - pk)/nsim
list(p_valueKolm=pvaluek, p_valueCramer=pvaluei)
}

```

ANEXO M

Programa simulaKSSI

```

function(event, nsim = 100, np=100){
#
# This function implements monte-carlo simulations
# for getting the power of a test against the hypothesis of complete
spatial
# randomness (CSR)of an observed spatial point pattern.
# Point patterns have area equal to one with 100 events (points)
#
# The test uses K-function (actually L-function) by Ripley.
# Cramer-vonMises and Kolmogorov measure of distances are used for
# getting statistics of tests.
#
# OBS: You need Spatstat and splancs!
#
# Where:
# nsim: number of monte-carlo simulations
# np: number of events
#
# This is crude code by J.D.Scalon. No warranty! (08/04/10)
#
# Get the distribution of the observed K-function
#
  r<-seq(0,0.25,length=200)
  k<- Kest(event, r=r, correction="Ripley")
  ko<-sqrt(k$iso/pi)
#
# Get the integral for the observed spatial point pattern
#
  lent <- length(r)
  D<-(ko-r)^2
  step<-max(r)/lent
  integ<-0
  steplength <- step * 2
  for(count in (2:(lent - 2))) {
    integ <- integ + (D[count - 1] + 4 * D[count] + D[count + 1]) *
      steplength
  }
  integ <- integ/6

```

```

#
# Get the kolmogorov distance for the observed spatial point pattern
#
  DK<-max(abs(ko-r))
#
# Doing Monte-Carlo simulation under the CSR
#
  for(i in (2:nsim)) {
    x<-runif(np,0,1)
    y<-runif(np,0,1)
    xy<-cbind(x,y)
    spts<- as.ppp(xy, c(0,1,0,1))
    kk<- Kest(spts,r=r, correction="Ripley")
    kl<-sqrt(kk$iso/pi)
#
# Get the integral under CSR
#
    lent <- length(r)
    t <- (kl - r)^2
    step<-max(r)/lent
    teg <- 0
    steplength <- step * 2
    for(count in (2:(lent - 2))) {
      teg <- teg + (t[count - 1] + 4 * t[count] + t[count + 1]) *
        steplength
    }
    teg <- teg/6
    integ <- c(integ, teg)
#
# Get kolmogorov distance under CSR
#
    test<-max(abs(kl-r))
    DK<-c(DK,test)
  }
pi <- rank(integ)[1]
pk <- rank(DK)[1]
pvaluei <- (nsim + 1 - pi)/nsim
pvaluek <- (nsim + 1 - pk)/nsim
list(p_valueKolm=pvaluek, p_valueCramer=pvaluei)
}

```

ANEXO N

```

Progrsms KENVELOPES
function(event, nsim = 100, ma=0.25, lengt=100){
#
# This function implements monte-carlo simulations
# to create and plot simulation envelopes to test
# against the hypothesis of complete spatatial randomness (CSR)
# of an observed spatial point pattern.
# It uses the area the Ripley´s L-function.
#
# OBS: You need Spatstat!
#
# Where:
# event: ppp class object that contains the coordinates of the events
# nsim: number of monte-carlo simulations
# lengt: number of classes of areas
# ma: Percentual value of the maximum size of area used in calculation
(0<ma<1)
#
# This is crude code by J.D.Scalon. No warranty! (25/05/10)
#
# starting values
#
  np<- length(event$x)
  ma<-ma*(max(event$x)*max(event$y))
#
# Get the L-function of the observed point pattern
#
  r<-seq(0,ma,length=lengt+1)
  rr<-seq(0,ma,length=lengt)
  k <- Kest(event, r=rr, correction="isotropic")
  ko<- sqrt(k$iso/pi)
#
#
# Doing Monte-Carlo simulation under the CSR
#
  hold<-matrix(0,nsim,lengt)
  for(i in (1:nsim)) {
    x<-runif(np,0,max(event$x))
    y<-runif(np,0,max(event$y))
    xy<-cbind(x,y)

```

```
spts<- as.ppp(xy, c(0,max(event$x),0,max(event$y)))
kk<- Kest(spts, r=rr, correction="isotropic")
kko<-sqrt(kk$iso/pi)
hold[i,]<-kko
}
#
# Observe that each row of hold contains the EDF for a single
# CSR simulation on the area
#
  up<- apply(hold,2,max)
  dow<- apply(hold,2,min)
#
#
# Creating plot
#
#
  y<-seq(0,max(ko),length=length(r))
par(pty = "s")
plot(r, y, type = "n", xlab = "Distances", ylab = "L-function")
lines(rr, ko, col="blue")
lines(rr, up, lty=2, col="red")
lines(rr, dow,lty=2,col="red")
}
```